



**Уральский
федеральный
университет**

имени первого Президента
России Б.Н.Ельцина

**Институт радиоэлектроники
и информационных
технологий**

**А. Г. ЗИНОВИК
Е. Э. СТРАШИНИН**

ИССЛЕДОВАНИЕ И СИНТЕЗ СИСТЕМЫ УПРАВЛЕНИЯ СТЕНДА «ТЕЛЕЖКА С ПОДВИЖНЫМ МАЯТНИКОМ»



Министерство образования и науки Российской Федерации
Уральский федеральный университет
имени первого Президента России Б. Н. Ельцина

А. Г. Зиновик, Е. Э. Страшинин

Исследование и синтез системы управления стенда «Тележка с подвижным маятником»

Рекомендовано методическим советом УрФУ
в качестве учебно-методического пособия
для студентов, обучающихся по направлению
220400 «Управление в технических системах»

Екатеринбург
Издательство Уральского университета
2014

УДК 681.5 (075.8)
ББК 32.973.26–018.2я73
З-63

Рецензенты:

зав. отделом динамических систем Института математики и механики УрО РАН,
член-корреспондент РАН, проф., д-р физ.-мат. наук **В. Н. Ушаков**
зав. каф. электротехники Уральского горного университета, доц., канд. техн. наук
Ф. Д. Босько

Зиновик, А. Г.

З-63 Исследование и синтез системы управления станда «Тележка с подвижным маятником»: учебно-методическое пособие / А. Г. Зиновик, Е. Э. Страшинин ; под общ. ред. Е. Э. Страшинина. — Екатеринбург : Изд-во Урал. ун-та, 2014. — 83, [1] с.

ISBN 978-5-7996-1285-6

Учебно-методическое пособие содержит описание выполнения четырех лабораторных работ по разделу «Линейные системы». Работы выполняются в пакете MATLAB с использованием станда «Тележка с подвижным маятником». Приведены описание станда, математическая модель объекта управления, требования к оформлению отчета и контрольные вопросы.

Библиогр.: 5 назв. Рис. 37. Табл. 14. Прил. 4.

УДК 681.5 (075.8)
ББК 32.973.26–018.2я73

ISBN 978-5-7996-1285-6

© Уральский федеральный университет, 2014

Оглавление

Введение	5
Лабораторная работа № 1. Моделирование объекта управления	6
1.1. Цели и задачи работы.....	6
1.2. Краткое описание стенда.....	7
1.3. Математическое описание объекта управления	11
1.3.1. Тележка с маятником	11
1.3.2. Двигатель в составе стенда	15
1.3.3. Линеаризация уравнений движения	17
1.4. Применение пакета MATLAB для моделирования объекта управления.....	19
1.4.1. Пакет расширения Symbolic Math Toolbox.....	19
1.4.2. Создание моделей в пакете расширения SIMULINK.....	20
1.5. Подготовительная часть.....	22
1.6. Выполнение работы	24
1.6.1. Создание личной папки для работы с моделями.....	24
1.6.2. Работа с векторно-матричными уравнениями линеаризованной системы	25
1.6.3. Работа с нелинейной моделью	25
1.6.4. Сравнение нелинейной и линеаризованной модели при свободных колебаниях маятника	29
1.7. Содержание отчета.....	31
Контрольные вопросы.....	32
Лабораторная работа № 2. Идентификация параметров объекта управления.....	33
2.1. Цели и задачи работы.....	33
2.2. Особенности работы со стендом в режиме реального времени	34
2.2.1. Реализация алгоритмов управления с учетом каналов связи с объектом	34
2.2.2. Порядок запуска системы управления в режиме реального времени.....	36
2.3. Выполнение работы	37
2.3.1. Создание личной папки для работы с моделями	37
2.3.2. Измерение сухого трения тележки	37
2.3.3. Формирование массива данных для проведения эксперимента по идентификации	38
2.3.4. Обработка массива данных перед проведением идентификации	41
2.3.5. Идентификация параметров объекта с помощью инструмента Control and Estimation Toolbox Manager	42
2.4. Содержание отчета.....	47
Контрольные вопросы.....	48

Лабораторная работа № 3. Модальный синтез управления в режиме крана	49
3.1. Цели и задачи работы	49
3.2. Подготовительная часть	50
3.3. Выполнение работы	52
3.3.1. Создание личной папки для работы с моделями	52
3.3.2. Нелинейная модель стенда для модального синтеза управления в режиме крана	52
3.3.3. Выбор желаемых собственных чисел матриц динамики замкнутой системы и наблюдателя	53
3.3.4. Порядок запуска системы управления в режиме реального времени	57
3.3.5. Проверка результатов модального синтеза на стенде	58
3.4. Содержание отчета	59
Контрольные вопросы	59
 Лабораторная работа № 4. Модальный синтез управления в режиме инвертированного маятника	60
4.1. Цели и задачи работы	60
4.2. Подготовительная часть	61
4.3. Выполнение работы	62
4.3.1. Создание личной папки для работы с моделями	62
4.3.2. Моделирование работы стенда в режиме инвертированного маятника	62
4.3.3. Порядок запуска системы управления в режиме реального времени	62
4.3.4. Проверка результатов модального синтеза управления инвертированным маятником на стенде	66
4.4. Содержание отчета	68
Контрольные вопросы	68
 Библиографический список	70
 Приложение 1. Тексты m-файлов, используемых в лабораторной работе № 1	71
Приложение 2. Тексты m-файлов, используемых в лабораторной работе № 2	75
Приложение 3. Тексты m-файлов, используемых в лабораторной работе № 3	78
Приложение 4. Тексты m-файлов, используемых в лабораторной работе № 4	82

Введение

Учебно-методическое пособие к лабораторным работам «Исследование и синтез системы управления стенда «Тележка с подвижным маятником» по дисциплине «Теория автоматического управления» предназначено для студентов направления 220400 «Управление в технических системах» очной формы обучения.

Изучение раздела «Линейные системы» дисциплины «Теория автоматического управления» предусматривает выполнение четырех лабораторных работ: моделирование объекта управления, идентификацию параметров объекта управления, модальный синтез управления в режиме крана, модальный синтез управления в режиме инвертированного маятника. Работы выполняются с использованием пакета MATLAB. Целью работ является демонстрация применения методов современной теории управления с использованием программно-инструментального комплекса «Тележка с подвижным маятником».

В разработанных лабораторных работах реализован современный подход к организации практической деятельности студентов. Студенты, выполняя лабораторные работы, проходят следующие основные этапы: разработку алгоритма, проверку и отладку алгоритма на модели, реализацию алгоритма на стенде в режиме реального времени. Таким образом, организовано непосредственное взаимодействие студента с реальным объектом управления, что является очень важным при подготовке квалифицированных специалистов в области систем автоматического управления.

Лабораторная работа № 1.

Моделирование объекта управления

1.1. Цели и задачи работы

Цель работы: изучить математическое описание объекта управления и провести моделирование нелинейной и линеаризованной систем.

Задачи:

- 1) ознакомиться с кратким описанием стенда;
- 2) изучить математическое описание объекта управления;
- 3) провести линеаризацию полной нелинейной модели и сравнить процессы в нелинейной и линеаризованной моделях.

Исходные данные указаны в табл. 1.1, 1.2.

Таблица 1.1

Перечень параметров объекта

Название параметра	Обозначение	Исходное значение	Единица измерения
Масса маятника	m	0,174	кг
Масса тележки	M	1,321	кг
Длина стержня маятника	L	0,38	м
Момент инерции стержня	J	0,00099	кг·м ²
Коэффициент вязкого трения тележки	b	0,05	Н·с / м
Коэффициент демпфирования маятника	d	0,005	Н·м·с
Гравитационная постоянная	g	9,81	м/с ²

Таблица 1.2

Перечень параметров двигателя

Название параметра	Обозначение	Исходное значение	Единица измерения
Конструктивная постоянная двигателя	$(C_m\Phi)_t = K_t$	0,05	Н·м/А
	$(C_m\Phi)_E = K_b$	0,05	В/(Рад/с)
Активное сопротивление обмотки ротора	R_a	2,5	Ом
Индуктивность обмотки ротора	L_a	0,0025	Гн
Радиус колеса (шкива)	R_k	0,025	м
Коэффициент вязкого трения на валу двигателя	$b_{дв}$	0,00568	Н·с·м

1.2. Краткое описание стенда

Стенд представляет собой программно-аппаратный комплекс Digital Pendulum, поставленный фирмой Feedback Instruments в Уральский федеральный университет (рис. 1.1). Комплекс предназначен для проведения научных исследований и учебного процесса в области систем автоматического управления. В состав комплекса входит подвижная тележка, сдвоенный маятник, двигатель постоянного тока, блок управления двигателем и интерфейсная плата, установленная в ПК.

Объект управления состоит из подвижной тележки с закрепленным на ней сдвоенным маятником. Маятник может свободно вращаться в вертикальной плоскости с неограниченным диапазоном угла вращения. Тележка передвигается по рельсу и приводится в движение двигателем постоянного тока через шкив с ременной передачей (рис. 1.2).

Напряжение, подаваемое на двигатель, является управляющим сигналом и находится в диапазоне от -24 В до 24 В. Перемещение тележки ограничивается конечными выключателями, которые прекращают подачу питания на двигатель, когда тележка наезжает на них. При продолжении движения тележки в сторону от центра она ударяется о специальные буферные ограничители.

Относительная позиция тележки и угловое положение маятника измеряются оптическими инкрементными кодовыми датчиками (см. рис. 1.2). Оптический кодовый датчик состоит из светодиодного источника с линзой, кодового диска и фотодетектора.

При прохождении кодового диска между светодиодным источником и фотодетектором датчик генерирует последовательный импульсный цифровой

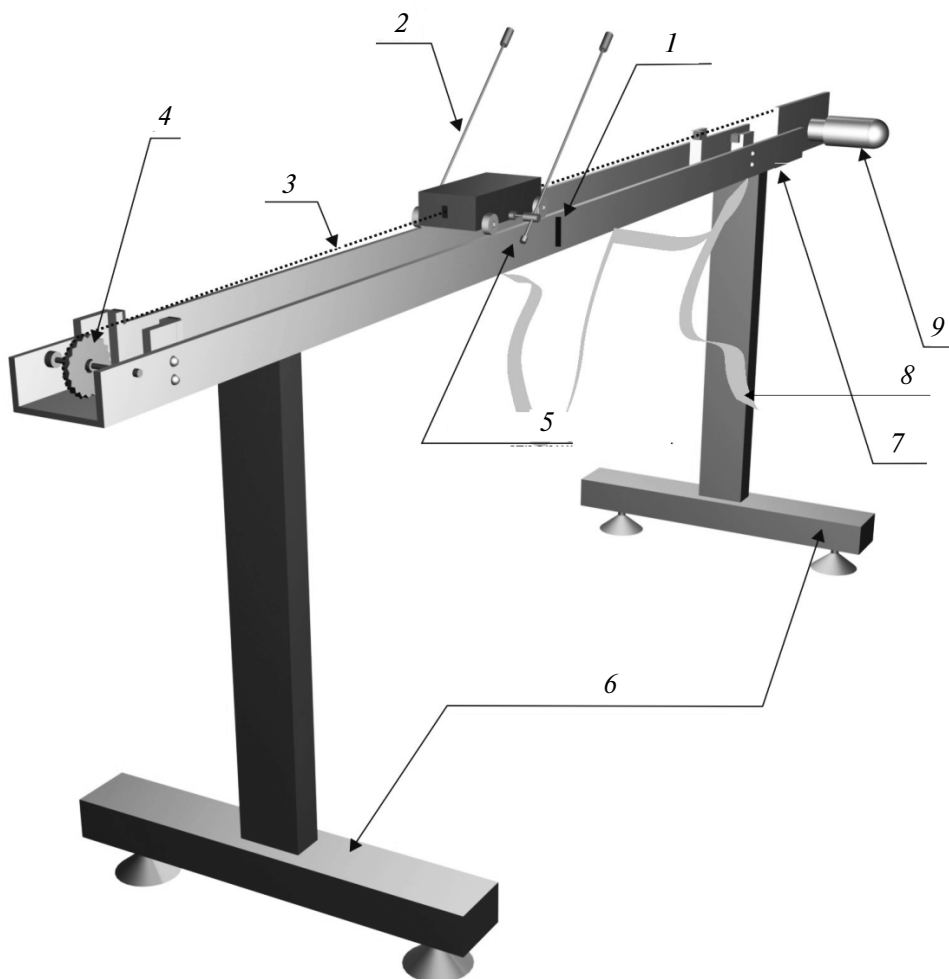


Рис. 1.1. Стенд со сдвоенным маятником:

1 — центральная метка, 2 — стержень с маятником, 3 — ремень, 4 — зубчатое колесо, 5 — винт для фиксации стержня, 6 — регулируемые стойки, 7 — регулировка натяжения ремня, 8 — ленточный кабель, 9 — двигатель постоянного тока

код, содержащий информацию относительно угла поворота маятника или позиции тележки.

Поскольку положение тележки определяется алгебраической суммой пришедших в компьютер импульсов, то перед каждым запуском системы тележка должна быть помещена в нулевую позицию (см. центральную метку на рис. 1.1). До начала запуска маятник должен быть успокоен и установлен в нижнее вертикальное положение, которому соответствует величина угла, равная π радиан.

Функциональная схема системы управления показана на рис. 1.3.

Информация с датчиков положения тележки и углового положения маятника подается в блок управления двигателем, она необходима для формирования управляющего воздействия. Сигналы с датчиков через интерфейс связи приходят на адаптер параллельного интерфейса SCSI, который, в свою

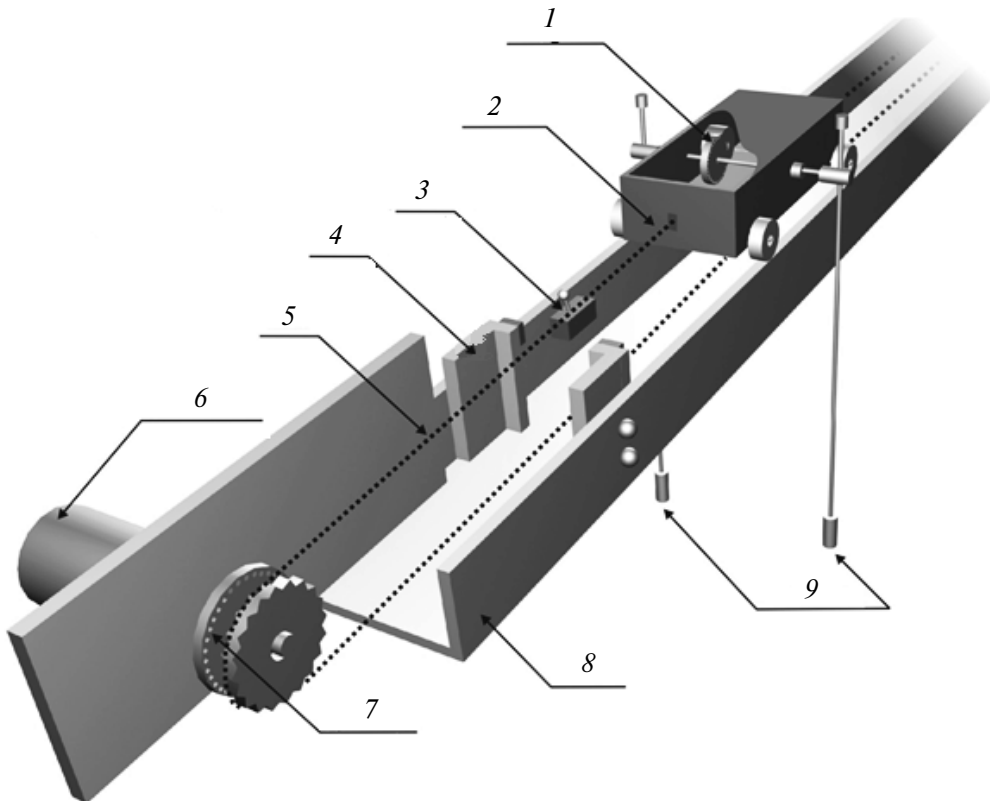


Рис. 1.2. Движение тележки по рельсу:

1 — Датчик углового положения маятника, 2 — тележка, 3 — концевой выключатель, 4 — буферный ограничитель, 5 — ремень, 6 — двигатель постоянного тока, 7 — датчик положения тележки, 8 — рельс, 9 — маятник

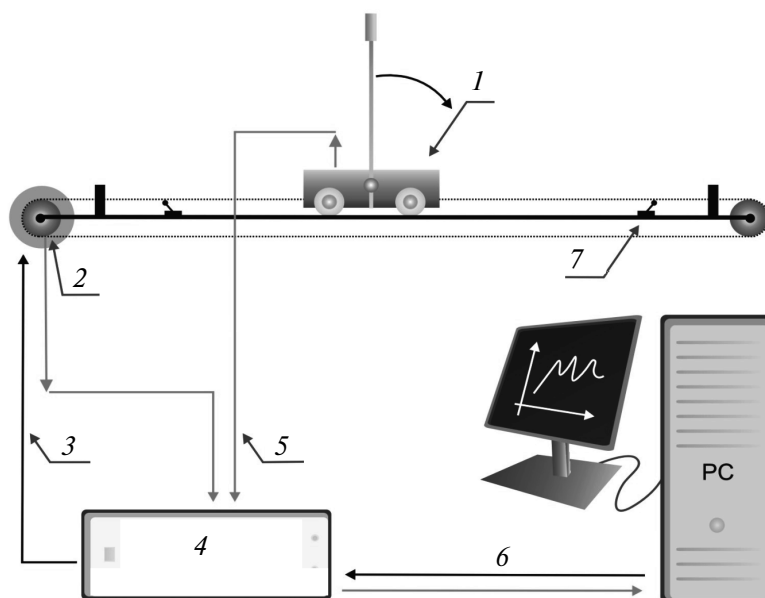


Рис. 1.3. Функциональная схема системы управления:

1 — тележка и угловое положение маятника, 2 — двигатель постоянного тока и датчик положения тележки, 3 — управление, 4 — блок управления двигателем и интерфейс связи с ПК, 5 — измеренное значение угла, 6 — алгоритмы управления, 7 — концевой выключатель

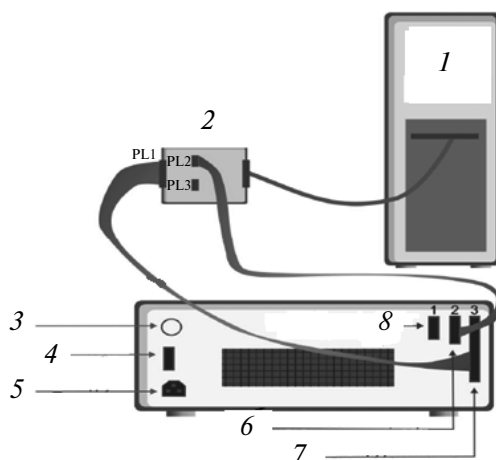


Рис. 1.4. Схема соединения блока управления двигателем с ПК через адаптер SCSI:

1 — персональный компьютер с интерфейсной платой PCI-1711, 2 — SCSI-адаптер, 3 — напряжение, подаваемое на двигатель, 4 — переключатель 230/110В, 5 — питание, 6 — шлейф для ПК (PL3 на адаптере SCSI), 7 — шлейф для ПК (PL1 на адаптере SCSI), 8 — информация с датчиков

очередь, отправляет данные в ПК. Схема соединения блока управления двигателем с ПК через адаптер SCSI показана на рис. 1.4. Для связи ПК и адаптера SCSI используется интерфейсная плата Advantech PCI-1711.

Сигналы с интерфейсной платы поступают в среду MATLAB Simulink.

1.3. Математическое описание объекта управления

1.3.1. Тележка с маятником

Рассмотрим тележку с закрепленным на ней маятником (рис. 1.5).

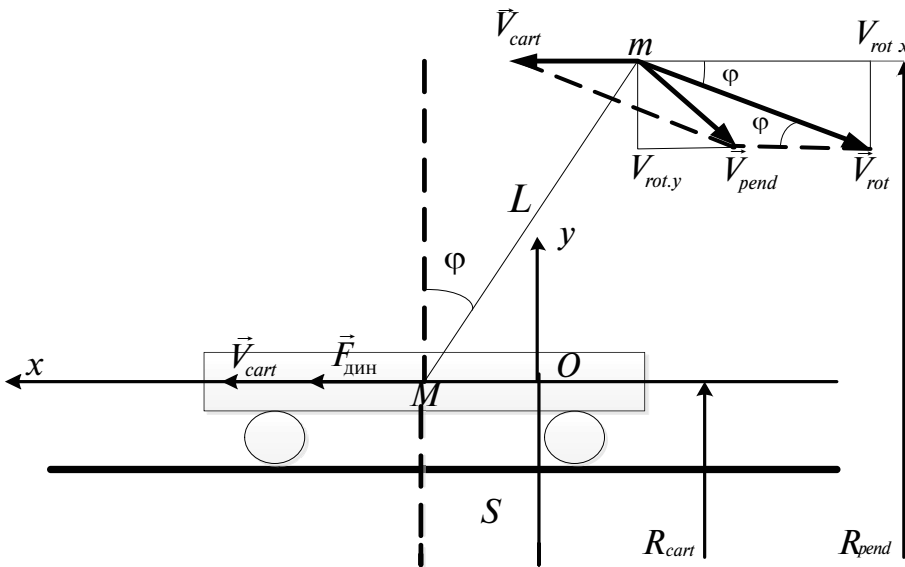


Рис. 1.5. Система маятника и тележки:

S — перемещение тележки; φ — угловое положение маятника; M — масса тележки; m — масса маятника; R_{cart} — расстояние от тележки до центра Земли; R_{pend} — расстояние от маятника до центра Земли; \vec{V}_{pend} — скорость маятника; \vec{V}_{rot} — вращательная составляющая скорости движения маятника; \vec{V}_{cart} — скорость тележки; $\vec{F}_{дин}$ — динамическая сила; L — длина стержня; $V_{rot.x}$ — проекция \vec{V}_{rot} на ось x ; $V_{rot.y}$ — проекция \vec{V}_{rot} на ось y

Для составления математического описания этой системы будем использовать уравнения Лагранжа [5]. Уравнения Лагранжа второго рода — это дифференциальные уравнения в обобщенных координатах, описывающие движение механической системы с n степенями свободы. Они находят широкое применение в исследовании поведения не только механических, но и других

физических систем (непрерывной среды, гравитационного или электромагнитного поля, электромеханических систем и др.).

Уравнение Лагранжа второго рода имеет следующий вид:

$$\frac{d}{dt} \left(\frac{\partial T(\dot{\vec{q}})}{\partial \dot{\vec{q}}} \right) - \frac{\partial T}{\partial \vec{q}} + \frac{\partial \Pi(\vec{q})}{\partial \vec{q}} + \frac{\partial D(\dot{\vec{q}})}{\partial \dot{\vec{q}}} = \vec{F},$$

где T — кинетическая энергия системы;

\vec{q} — вектор обобщенных координат;

Π — потенциальная энергия системы;

D — диссипативная энергия системы;

\vec{F} — вектор обобщенных сил;

Для i -ой степени свободы уравнение выглядит следующим образом:

$$\frac{d}{dt} \left(\frac{\partial T(\dot{\vec{q}})}{\partial \dot{q}_i} \right) - \frac{\partial T}{\partial q_i} + \frac{\partial \Pi(\vec{q})}{\partial q_i} + \frac{\partial D(\dot{\vec{q}})}{\partial \dot{q}_i} = \bar{F}_i, \quad (1.1)$$

где q_i — обобщенные координаты;

\bar{F}_i — обобщенные силы;

$i = 1, 2, 3 \dots n$ — номера степеней свободы.

Выберем в качестве обобщенных координат $q_1 = S$ и $q_2 = \varphi$, в этом случае обобщенные силы — $\bar{F}_1 = F_{\text{дин}}$, Н, $\bar{F}_2 = 0$.

Кинетическая энергия системы равна сумме кинетических энергий маятника и тележки: $T = T_{\text{cart}} + T_{\text{pend}}$, где T_{cart} — кинетическая энергия тележки, T_{pend} — кинетическая энергия маятника.

Кинетическая энергия тележки вычисляется по формуле:

$$T_{\text{cart}} = \frac{M\dot{S}^2}{2},$$

где M — масса тележки, кг

Кинетическая энергия маятника

$$T_{\text{pend}} = \frac{mV_{\text{pend}}^2}{2}, \quad (1.2)$$

где m — масса маятника, кг.

Скорость маятника можно представить в виде геометрической суммы скоростей вращательного движения маятника вокруг оси O и поступательного движения (движения тележки):

$$\vec{V}_{\text{pend}} = \vec{V}_{\text{rot}} + \vec{V}_{\text{cart}}.$$

Спроектируем уравнение на горизонтальную ось X (рис. 1.5).

Согласно этому рисунку горизонтальная составляющая скорости маятника может быть представлена в виде:

$$V_{pend.x} = V_{cart} - V_{rot} \cos \varphi. \quad (1.3)$$

Вертикальная составляющая скорости маятника:

$$V_{pend.y} = -V_{rot} \sin \varphi. \quad (1.4)$$

Квадрат скорости маятника равен сумме квадратов горизонтальной (1.3) и вертикальной (1.4) составляющих:

$$V_{pend}^2 = V_{pend.x}^2 + V_{pend.y}^2 = V_{cart}^2 - 2V_{cart}V_{rot} \cos \varphi + V_{rot}^2 \cos^2 \varphi + V_{rot}^2 \sin^2 \varphi. \quad (1.5)$$

Упростив (1.5), получим:

$$V_{pend}^2 = V_{cart}^2 - 2V_{cart}V_{rot} \cos \varphi + V_{rot}^2. \quad (1.6)$$

Скорость тележки равна производной от перемещения тележки:

$$V_{cart} = \dot{S}. \quad (1.7)$$

Вращательная скорость маятника равна произведению угловой скорости $\omega = \frac{d\varphi}{dt}$ на длину маятника L :

$$V_{rot} = \omega L = L\dot{\varphi}. \quad (1.8)$$

Подставив (1.7, 1.8) в (1.6), получим:

$$V_{pend}^2 = \dot{S}^2 - 2\dot{S}L\dot{\varphi} \cos \varphi + L^2\dot{\varphi}^2. \quad (1.9)$$

Подставим (1.9) в (1.2):

$$T_{pend} = \frac{m\dot{S}^2 - 2mL\dot{S}\dot{\varphi} \cos \varphi + mL^2\dot{\varphi}^2}{2}.$$

Суммарная кинетическая энергия равна:

$$T = \frac{(m+M)\dot{S}^2 - 2mL\dot{S}\dot{\varphi} \cos \varphi + mL^2\dot{\varphi}^2}{2}.$$

Потенциальная энергия системы равна сумме потенциальных энергий маятника и тележки:

$$\Pi = \Pi_{cart} + \Pi_{pend},$$

$$\Pi_{cart} = MgR_{cart},$$

где g — гравитационная постоянная.

Потенциальная энергия маятника:

$$\Pi_{pend} = mgR_{pend} = mg(R_{cart} + L \cos \varphi).$$

Вычислим частную производную кинетической энергии по \dot{S} :

$$\frac{\partial T}{\partial \dot{S}} = (M + m)\dot{S} - mL\dot{\varphi} \cos \varphi. \quad (1.10)$$

Продифференцируем по времени (1.10):

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{S}} \right) = (M + m)\ddot{S} - mL\ddot{\varphi} \cos \varphi + mL\dot{\varphi}^2 \sin \varphi.$$

Частные производные от кинетической и потенциальной энергии по S :

$$\frac{\partial T}{\partial S} = 0; \quad \frac{\partial \Pi}{\partial S} = 0.$$

Вычислим диссипативную энергию системы:

$$D = \frac{bV_{cart}^2}{2} + \frac{dV_{pend}^2}{2}.$$

Частные производные от диссипативной энергии:

$$\frac{\partial D}{\partial \dot{S}} = bV_{cart}; \quad \frac{\partial D}{\partial \dot{\varphi}} = d\omega_{pend}.$$

В соответствии с (1.1) запишем уравнение Лагранжа для первой обобщенной координаты S :

$$(M + m)\ddot{S} - mL\ddot{\varphi} \cos \varphi + mL\dot{\varphi}^2 \sin \varphi + b\dot{S} = F_{дин}.$$

Рассмотрим уравнения для второй обобщенной координаты:

$$\frac{\partial T}{\partial \dot{\varphi}} = -mL\dot{S} \cos \varphi + mL^2\dot{\varphi}. \quad (1.11)$$

Производная по времени от (1.11):

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\varphi}} \right) = -mL\ddot{S} \cos \varphi + mL\dot{S} \sin \varphi + mL^2\ddot{\varphi}$$

Частные производные от кинетической и потенциальной энергии:

$$\frac{\partial T}{\partial \dot{\varphi}} = mL\dot{S}\dot{\varphi} \sin \varphi; \quad \frac{\partial \Pi}{\partial \varphi} = -mLg \sin \varphi.$$

Запишем уравнения для второй обобщенной координаты φ :

$$mL^2\ddot{\varphi} - mL\ddot{S} \cos \varphi - mgL \sin \varphi + d\dot{\varphi} = 0.$$

При выводе этого уравнения полагалось, что масса маятника сосредоточена в одной точке, и сам маятник подвешен на невесомой нити. Учет массы стержня, на котором закреплен маятник, в соответствии с правилами теоретической механики приводит к уравнению:

$$(J + mL^2)\ddot{\varphi} - mL\ddot{S} \cos \varphi - mgL \sin \varphi + d\dot{\varphi} = 0,$$

где J — момент инерции стержня.

Таким образом, получена система уравнений Лагранжа второго порядка:

$$\begin{cases} (M + m)\ddot{S} + b\dot{S} - mL\ddot{\varphi} \cos \varphi + mL\dot{\varphi}^2 \sin \varphi = F_{\text{дин}}; \\ (J + mL^2)\ddot{\varphi} + d\dot{\varphi} - mgL \sin \varphi - mL\ddot{S} \cos \varphi = 0. \end{cases} \quad (1.12)$$

1.3.2. Двигатель в составе стенда

Для нахождения силы $F_{\text{дин}}$, входящей в систему уравнений (1.12), рассмотрим математическую модель двигателя с кинематической схемой, представленной на рис. 1.6.

Для перемещения тележки по рельсу используется двигатель постоянного тока с независимым возбуждением.

При положительном напряжении, подаваемом на якорь двигателя, его вал вращается в направлении, показанном на рис. 1.6. Напряжение, подаваемое на обмотку якоря двигателя, уравнивается падением напряжения на активном сопротивлении обмотки якоря, ЭДС самоиндукции якоря и ЭДС вращения:

$$U_{\text{дв}} = R_a i_a + L_a \frac{di_a}{dt} + E_{\text{дв}}, \quad (1.13)$$

где i_a — сила тока якоря;

R_a — сопротивления обмотки якоря;

L_a — индуктивность обмотки якоря;

$E_{\text{дв}}$ — ЭДС вращения ротора двигателя.

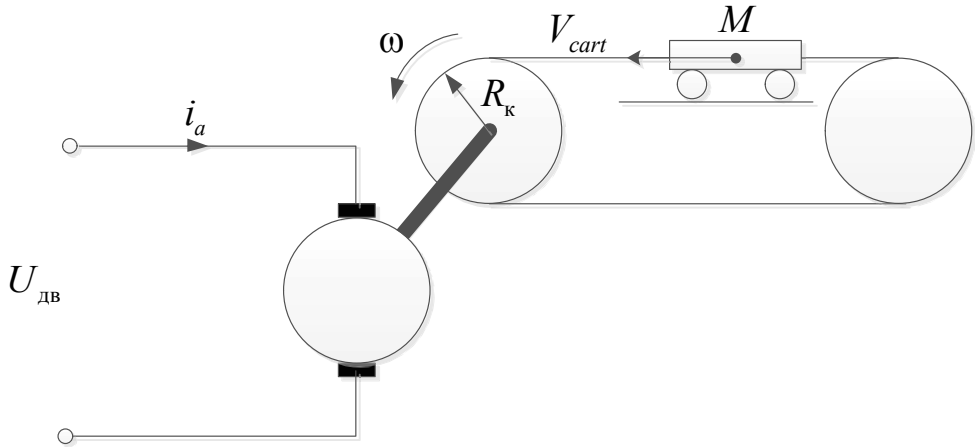


Рис. 1.6. Двигатель постоянного тока:

ω — угловая скорость двигателя, R_k — радиус зубчатого колеса, закрепленного на роторе двигателя, i_a — сила тока якоря, M — масса тележки, V_{cart} — скорость тележки

ЭДС вращения можно выразить следующей формулой:

$$E_{\text{дв}} = C_m \Phi \cdot \omega_{\text{дв}}, \quad (1.14)$$

где $C_m \Phi$ — коэффициент, связанный с конструктивными параметрами двигателя, обозначим этот коэффициент K_b .

Подставив (1.14) в (1.13), получим:

$$U_{\text{дв}} = R_a i_a + L_a \frac{di_a}{dt} + K_b \omega_{\text{дв}}. \quad (1.15)$$

В соответствии с рис. 1.7 угловая скорость вала двигателя связана с линейной скоростью перемещения тележки соотношением:

$$\omega_{\text{дв}} = \frac{V_{\text{cart}}}{R_k}.$$

Разрешим (1.15) относительно $\frac{di_a}{dt}$:

$$\frac{di_a}{dt} = \frac{1}{L_a} \left(U_{\text{дв}} - R_a i_a - \frac{K_b}{R_k} V_{\text{cart}} \right).$$

Электромагнитный момент двигателя пропорционален току якоря:

$$M_{\text{эм}} = C_m \Phi \cdot i_a.$$

В данном случае используется обозначение $C_m \Phi = K_t$. Если все физические величины используются в системе СИ, то $K_b = K_t$.

Учтем трение на валу двигателя, пропорциональное, с коэффициентом $b_{дв}$, угловой скорости вращения вала (так называемое «вязкое трение»). Результирующий момент:

$$M_{дв} = M_{эм} - M_{тр}^{вязк},$$

момент вязкого трения

$$M_{тр}^{вязк} = b_{дв} \cdot \omega_{дв}.$$

Ввиду малости момента инерции ротора двигателя, им можно пренебречь. Электромагнитный момент двигателя через шкив радиуса R_k формирует динамическую силу, приводящую в движение тележку:

$$F_{дин} = \frac{M_{дв}}{R_k} = (M_{эм} - M_{тр}^{вязк}) / R_k = (K_t \cdot i_a - b_{дв} \cdot \omega_{дв}) / R_k. \quad (1.16)$$

Таким образом, математическую модель двигателя можно представить в виде структурной схемы, представленной на рис. 1.7.

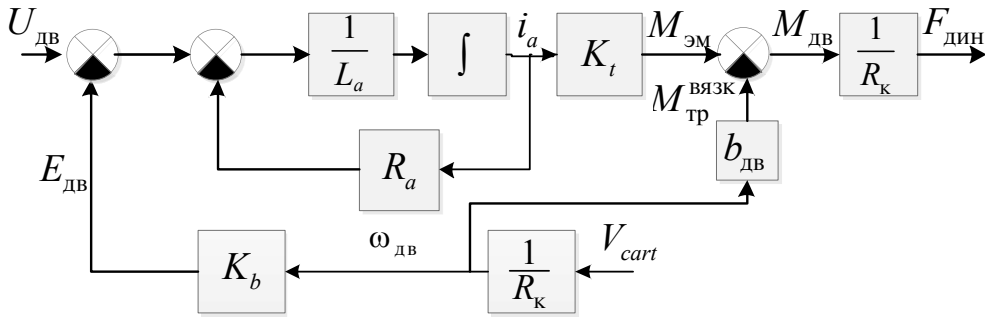


Рис. 1.7. Структурная схема модели двигателя

1.3.3. Линеаризация уравнений движения

Рассмотрим систему уравнений (1.12) и проведем их линеаризацию. Будем полагать, что движение по каждой из координат представлено в виде суммы базового (базовой точки) и приращения относительно базового:

$$S = S_0 + \Delta S; \quad \varphi = \varphi_0 + \Delta \varphi; \quad S_0 = \text{const}; \quad \varphi_0 = \text{const}.$$

Проведем линеаризацию уравнения:

$$(J + mL^2)\ddot{\varphi} = mL\ddot{S}\cos\varphi + mgL\sin\varphi - d\dot{\varphi}, \quad (1.17)$$

учитывая приведенные выше допущения. Для этого разложим каждую нелинейную функцию, входящую в (1.17), в ряд Тейлора:

$$\begin{aligned} (J + mL^2)(\varphi_0 + \Delta\varphi)'' &= mL\ddot{S}\cos\varphi_0 + \left. \frac{\partial(mL\ddot{S}\cos\varphi)}{\partial\varphi} \right|_{\varphi_0} \Delta\varphi + \\ &+ mgL\sin\varphi_0 + \left. \frac{\partial(mgL\sin\varphi)}{\partial\varphi} \right|_{\varphi_0} \Delta\varphi + R - d(\varphi_0 + \Delta\varphi)', \end{aligned} \quad (1.18)$$

где R — остаточный член ряда Тейлора, которым можно пренебречь.

Упростив (1.18), получим:

$$(J + mL^2)\Delta\ddot{\varphi} = mL\ddot{S} + mgL\Delta\varphi - d\Delta\dot{\varphi}.$$

Аналогично, проводя линеаризацию уравнения

$$(M + m)\ddot{S} = F_{\text{дин}} + mL\ddot{\varphi}\cos\varphi - mL\dot{\varphi}^2\sin\varphi - b\dot{S},$$

получим:

$$(M + m)\Delta\ddot{S} = F_{\text{дин}} + mL\Delta\ddot{\varphi} - b\Delta\dot{S}.$$

В дальнейшем символом Δ будем пренебрегать. Таким образом, система линеаризованных уравнений при $\varphi_0 = 0$ выглядит следующим образом:

$$\begin{cases} (m + M)\ddot{S} + b\dot{S} - mL\ddot{\varphi} = F_{\text{дин}}; \\ (J + mL^2)\ddot{\varphi} + d\dot{\varphi} - mLg\varphi - mL\ddot{S} = 0. \end{cases} \quad (1.19)$$

Аналогично для нижнего вертикального положения маятника, при $\varphi_0 = \pi$, проводя линеаризацию уравнений (1.12), получим:

$$\begin{cases} (m + M)\ddot{S} + b\dot{S} + mL\ddot{\varphi} = F_{\text{дин}}; \\ (J + mL^2)\ddot{\varphi} + d\dot{\varphi} + mLg\varphi + mL\ddot{S} = 0. \end{cases} \quad (1.20)$$

1.4. Применение пакета MATLAB для моделирования объекта управления

1.4.1. Пакет расширения Symbolic Math Toolbox

Symbolic Math Toolbox — это пакет расширения MATLAB, содержащий функции аналитических преобразований и поддерживающий арифметику произвольной точности.

Возможности пакета Symbolic Math Toolbox:

- аналитические вычисления (символьное интегрирование, дифференцирование, упрощение, преобразование математических выражений и решение уравнений);
- реализация арифметических операций с произвольной точностью;
- использование языка MuPAD для работы над символьными математическими выражениями;
- конвертирование символьных выражений в MATLAB, C, Fortran, MathML и TeX;
- создание пользовательских символьных функций и библиотек.

Приведем основные функции пакета Symbolic Math Toolbox, используемые в лабораторной работе.

Функция установки количества знаков чисел digits

Арифметикой произвольной точности называют вычисления, у которых все числа результатов являются точными. Функция `digits` служит для установки числа цифр в числах арифметики произвольной точности. Она используется в следующем виде:

- `digits` — возвращает число значащих цифр в числах арифметики произвольной точности (по умолчанию 32);
- `digits (D)` — устанавливает заданное число цифр `D` для чисел арифметики произвольной точности (`D` — целое число).

Функция для создания символьных переменных sym

`S = sym (A)` — возвращает символьный объект `S` класса 'sym' для входного параметра `A`. Если `A` — строка, то будет получена символьная строка или символьная переменная, а если `A` — это число (скаляр) или матрица, то будут получены их символьные представления.

Для преобразования чисел или матриц в символьную форму или для присваивания значений символьным переменным применяется функция `sym` в виде:

`S = sym (A, flag)`, где `A` — это число (скаляр) или матрица.

Второй аргумент — `flag` — задается в следующем виде:

- ‘f’ — число с плавающей точкой;
- ‘r’ — число в рациональной форме (задано по умолчанию);
- ‘e’ — число в рациональной форме плюс оценка машинной погрешности;
- ‘d’ — число в расширенной десятичной форме с числом верных цифр, заданных функцией `digits`.

Для создания группы символьных объектов служит функция `syms`. Пример: `syms arg1 arg2` — создает группу символьных объектов `arg1 arg2`.

Функция вывода символьных выражений `pretty`

`pretty (S)` — выводит выражение `S` в формате, приближенном к математическому.

Функция обеспечения подстановок `subs`

`Subs (S)` — заменяет в символьном выражении `S` все переменные их символьными значениями, которые берутся из вычисляемой функции или рабочей области системы `MATLAB`.

1.4.2. Создание моделей в пакете расширения `SIMULINK`

`SIMULINK` — пакет для моделирования и анализа динамических систем. `SIMULINK` позволяет эффективно изучать разнообразные системы из технических, физических и иных приложений, рассматривать нелинейные задачи с непрерывным и дискретным временем. Среда `SIMULINK` обладает открытой архитектурой и интегрирована в `MATLAB`, что позволяет использовать графику и библиотеки `MATLAB` при исследовании моделей.

Графический интерфейс `SIMULINK` используется на всех стадиях моделирования и значительно облегчает работу. Этап подготовки модели заключается в сборке схемы процесса из готовых, заложенных в `SIMULINK` элементов и настройке параметров `SIMULINK` системы. Чтобы упростить подготовку модели, применяются блочные диаграммы.





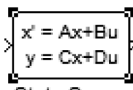
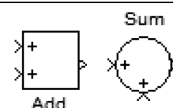
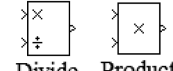



Для введенной модели можно проводить собственно симуляцию, исследовать равновесия и выводить результаты в графическом и цифровом виде. Можно создавать новые элементы и библиотеки элементов, группировать их в блоки и создавать иерархические модели.

В `SIMULINK` имеется обширная библиотека элементов (блоков) для анализа линейных и нелинейных систем, дискретных, непрерывных и гибридных процессов, ММО- и SISO-моделей. Допускается иерархическая структуризация моделей неограниченной вложенности. Для решения систем дифференциальных уравнений имеется несколько интеграторов, специальный акселератор для ускорения расчетов, можно также создавать и компилировать модули на языке `C` [1].

Основные элементы библиотеки компонентов, используемые в лабораторной работе, приведены в табл. 1.3

Таблица 1.3

Элементы библиотеки компонентов, используемые в лабораторной работе

Графическое изображение элемента	Расположение элемента	Описание элемента
 Constant	Simulink/Commonly Used Block	Задаёт постоянную величину (константу)
 Gain	Simulink/Commonly Used Block	Входной сигнал умножается на постоянную величину, заданную в свойствах блока
 In1 Out1	Simulink/Commonly Used Block	Входные и выходные порты модели
 Integrator	Simulink/Continuous	Интегрирование входного сигнала
 State-Space	Simulink/Continuous	Модель системы в пространстве состояний (State-space model)
 Add Sum	Simulink/Math Operations	Суммирование или вычитание входных сигналов
 Divide Product	Simulink/Math Operations	Умножение или деление входных сигналов
 Math Function	Simulink/Math Operations	В блоке реализованы различные математические функции: логарифмические, экспоненциальные, возведение в степень, извлечение корня и т.д.
 Trigonometric Function	Simulink/Math Operations	В блоке реализованы различные тригонометрические функции: синус, косинус, тангенс и т.д.
 Step	Simulink/Sources	Формирование ступенчатого воздействия

Окончание табл. 1.3

Графическое изображение элемента	Расположение элемента	Описание элемента
 Sine Wave	Simulink/Sources	Формирование синусоидального сигнала с заданными параметрами
 Manual Switch	Simulink/Signal Routing	Переключатель между двумя входными сигналами
 Demux Mux	Simulink/Signal Routing	Объединение или разделение входных сигналов
 Scope	Simulink/Sinks	Осциллограф. Используется для визуализации (просмотра) сигналов
 simout To Workspace	Simulink/Sinks	Запись входного сигнала в рабочую область

1.5. Подготовительная часть

1. Требуется представить систему нелинейных уравнений (1.12) в виде структурной схемы для последующего моделирования в пакете MATLAB. В качестве примера в разделе 1.3.2 на рис. 1.7 показана структурная схема модели двигателя, которая была составлена по уравнениям (1.15), (1.16).

Составленную структурную схему нелинейных уравнений (1.12) необходимо сдать на проверку преподавателю перед выполнением работы.

2. Требуется составить векторно-матричное описание линеаризованной системы. Для этого необходимо преобразовать линеаризованные системы уравнений (1.19, 2.20) к векторно-матричным дифференциальным уравнениям в форме Коши:

$$\begin{aligned}\dot{\bar{\mathbf{x}}}(t) &= \mathbf{A}\bar{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t); \\ \bar{\mathbf{y}}(t) &= \mathbf{C}\bar{\mathbf{x}}(t) + \mathbf{D}\mathbf{u}(t);\end{aligned}$$

где $\bar{\mathbf{x}}(t)$ — вектор состояния,

\mathbf{A} — матрица динамики,

\mathbf{B} — матрица входа,

$\mathbf{u}(t)$ — управление,
 $\bar{\mathbf{y}}(t)$ — вектор выхода,
 \mathbf{C} — матрица выхода,
 \mathbf{D} — матрица обхода.

Для получения векторно-матричных уравнений из системы (1.19) требуется составить систему уравнений вида:

$$\begin{cases} \dot{x}_1 = a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + a_{14}x_4 + a_{15}x_5 + b_1u_{\text{дв}} \\ \dot{x}_2 = a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + a_{24}x_4 + a_{25}x_5 + b_2u_{\text{дв}} \\ \dot{x}_3 = a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + a_{34}x_4 + a_{35}x_5 + b_3u_{\text{дв}} \\ \dot{x}_4 = a_{41}x_1 + a_{42}x_2 + a_{43}x_3 + a_{44}x_4 + a_{45}x_5 + b_4u_{\text{дв}} \\ \dot{x}_5 = a_{51}x_1 + a_{52}x_2 + a_{53}x_3 + a_{54}x_4 + a_{55}x_5 + b_5u_{\text{дв}} \end{cases} \quad (1.21)$$

где $a_{11} \div a_{55}$ — элементы матрицы \mathbf{A} ,

$b_1 \div b_5$ — элементы матрицы \mathbf{B} ,

$u_{\text{дв}}$ — управляющее напряжение двигателя.

Для этого выполните следующие действия:

1) введите вектор состояния $\bar{\mathbf{x}}$ и управление \mathbf{u} :

$$\bar{\mathbf{x}} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix}, \quad \mathbf{u} = u_{\text{дв}},$$

где $x_1 = i_a$, $x_2 = S$, $x_3 = \dot{S}$, $x_4 = \varphi$, $x_5 = \dot{\varphi}$,

2) замените переменные в системе уравнений (1.19) на координаты вектора состояния. Заметьте, что $\dot{x}_2 = \dot{S}$, $\dot{x}_4 = \dot{\varphi}$, поэтому $\dot{x}_2 = x_3$, $\dot{x}_4 = x_5$. Преобразуйте систему (1.19) в систему вида (1.21). Заметьте, что уравнения для \dot{x}_1 можно получить из уравнения двигателя (1.16), в котором $V_{\text{card}} = \dot{S} = x_3$. Уравнение для \dot{x}_3 и \dot{x}_5 можно получить из первого и второго уравнений системы (1.19).

В полученной системе уравнений вида (1.21) в правой части не должно остаться производных от координат вектора состояния.

Получив систему уравнений (1.21), необходимо записать матрицы \mathbf{A} и \mathbf{B} . Элементы матриц \mathbf{A} и \mathbf{B} должны быть представлены в символьной форме, с использованием обозначения параметров объекта и двигателя, приведенных в табл. 1.1, 1.2.

Выходные координаты системы — S и φ , поэтому матрица \mathbf{C} будет иметь вид:

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix},$$

матрица $D_{2 \times 1}$ — нулевая.

Аналогично требуется получить систему векторно-матричных уравнений вида (1.21) для системы уравнений (1.20) и записать полученные матрицы A и B в символьной форме.

Полученные векторно-матричные уравнения для систем линеаризованных уравнений (1.19) и (1.20) необходимо сдать на проверку преподавателю перед выполнением работы.

1.6. Выполнение работы

1.6.1. Создание личной папки для работы с моделями

На диске D:/ создайте папку с идентификатором: **ЛР1_Фамилия_Группа_Год**. Скопируйте в эту папку m-файлы и mdl-файлы, полученные от преподавателя. Запустите MATLAB и перейдите в созданную папку с файлами. Для этого в окне Current folder введите следующий путь: **D:/Имя созданной папки**.

1.6.2. Работа с векторно-матричными уравнениями линеаризованной системы

Используя полученные в ходе предварительной подготовки векторно-матричные уравнения линеаризованной системы, с помощью пакета MATLAB требуется вычислить передаточные функции для нижнего и верхнего вертикального положения маятника и построить графики весовых и переходных функций, ЛАЧХ, ЛФЧХ и АФХ.

Для работы с векторно-матричными уравнениями откройте m-файл lab1.m. Содержание m-файла приведено в прил. 1.

В m-файле lab1.m производится вывод векторно-матричных уравнений в символьческой форме, с использованием возможностей пакета Symbolic Toolbox. Далее осуществляется подстановка параметров объекта управления и двигателя, преобразование символьных матриц в числовой формат, создание векторно-матричного описания системы, вычисление матричной передаточной функции и вывод графиков с помощью LTI Viewer (средства графического интерфейса пользователя, используемого при расчете и анализе lti-систем).


При выполнении команды `ltiview` появляется графическое окно, в котором отображается график указанной характеристики. Дополнительные настройки доступны при нажатии правой клавиши мыши и через команды меню. Необходимой настройкой является включение сетки — `grid`, а также отключение опции `Show\Negative frequency` (отрицательные частоты) на графике АФХ.

Для сохранения содержимого графического окна `LTI Viewer` необходимо воспользоваться командой меню `File\Print to Figure`, после чего открывается новое окно `Figure`, содержащее тот же самый рисунок, скопировать который можно путем выполнения команды меню `Edit\Copy Figure`.

Требуется провести запуск `m`-файла `lab1.m` для верхнего (`select=0`) и нижнего (`select=1`) вертикального положения. После каждого запуска `m`-файла, перейдя в командное окно, скопируйте в отчет матрицы **A**, **B**, **C**, **D** с подставленными значениями исходных данных. Также сохраните графики весовых и переходных функций, ЛАЧХ и ЛФЧХ, АФХ, которые отобразились после запуска `m`-файла.

1.6.3. Работа с нелинейной моделью

Откройте файл `lab1_model1.mdl`, полученный от преподавателя (рис. 1.8). В данном файле требуется реализовать нелинейную модель объекта. Блок «Объект управления» состоит из двух подсистем: «Модель двигателя постоянного тока» и «Модель маятника с тележкой» (рис. 1.9).

Перейдите в подсистему «Модель двигателя постоянного тока» и создайте в ней структурную схему модели двигателя, приведенную на рис 1.7 (п. 1.3.2). При создании структурной схемы необходимо использовать элементы библиотеки компонентов `Simulink`, указанные в табл. 1.3 (п. 1.4.2). Для открытия библиотеки компонентов необходимо выбрать в меню `View` пункт `Library Browser` или нажать на пиктограмму  на панели инструментов.

Аналогично в подсистеме «Модель маятника с тележкой» создайте из элементов библиотеки компонентов структурную схему системы нелинейных уравнений (1.12), которая была получена в подготовительной части. В создаваемой структурной схеме необходимо предусмотреть возможность задания начального углового положения маятника. Для этого в свойствах блока «Интегратор», выходом которого является угловое положение маятника, необходимо указать переменную, задающую начальное угловое положение маятника.

При создании структурных схем модели двигателя и нелинейных уравнений объекта требуется присвоить номера интеграторам в соответствии с номерами координат вектора состояния. Интегратору, выходом которого является ток якоря i_a , требуется присвоить номер 1, интегратору, выходом которого является перемещение тележки S , — номер 2 и т.д.

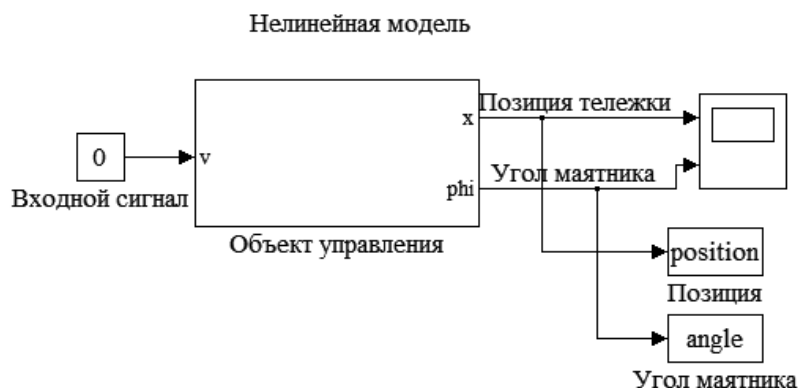


Рис. 1.8. Нелинейная модель

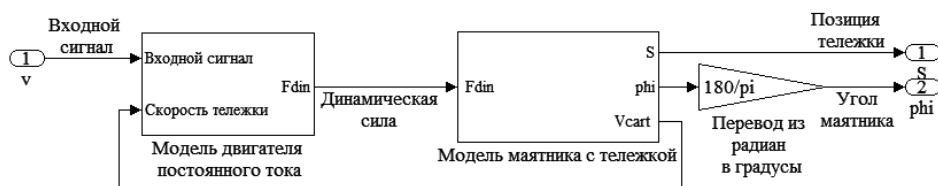


Рис. 1.9. Блок «Объект управления»

Для корректной работы модели требуется перед ее запуском загрузить в рабочую область MATLAB численные значения параметров, используемых в модели. Для этого рекомендуется создать m-файл `lab1_model1_params.m` с численными значениями параметров и начальным условием для углового положения маятника. Пример m-файла приведен в прил. 1. Сохраните созданный m-файл в рабочей папке.

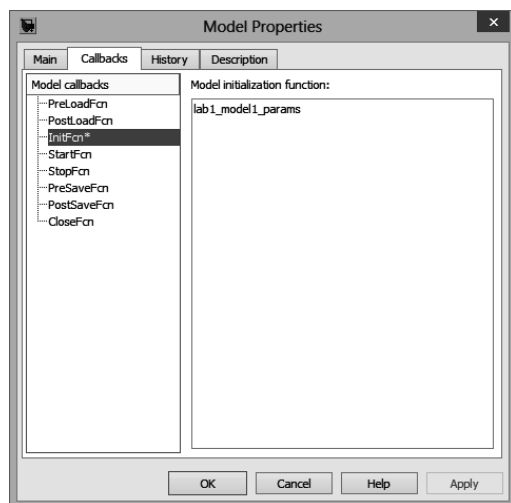


Рис. 1.10. Вкладка Model Properties

Подключите созданный m-файл к модели `lab1_model1.mdl`. Для этого в модели через меню `File\Model Properties` откройте вкладку «Свойства модели» (Model Properties). Перейдите на вкладку `Callbacks` и в дереве `Model Callbacks` выберите пункт `InitFcn`. В окне справа укажите названия m-файла, в котором записаны параметры модели (рис. 1.10).

Требуется провести линеаризацию созданной модели с помощью

инструмента Control and Estimation Tools. Для этого на входном сигнале, идущем в блок «Объект управления» с двигателем постоянного тока, нажав правую кнопку мыши, установите точку Input point, как показано на рис. 1.11. На выходных сигналах «Позиция» и «Угол маятника» аналогично установите точки Output point.

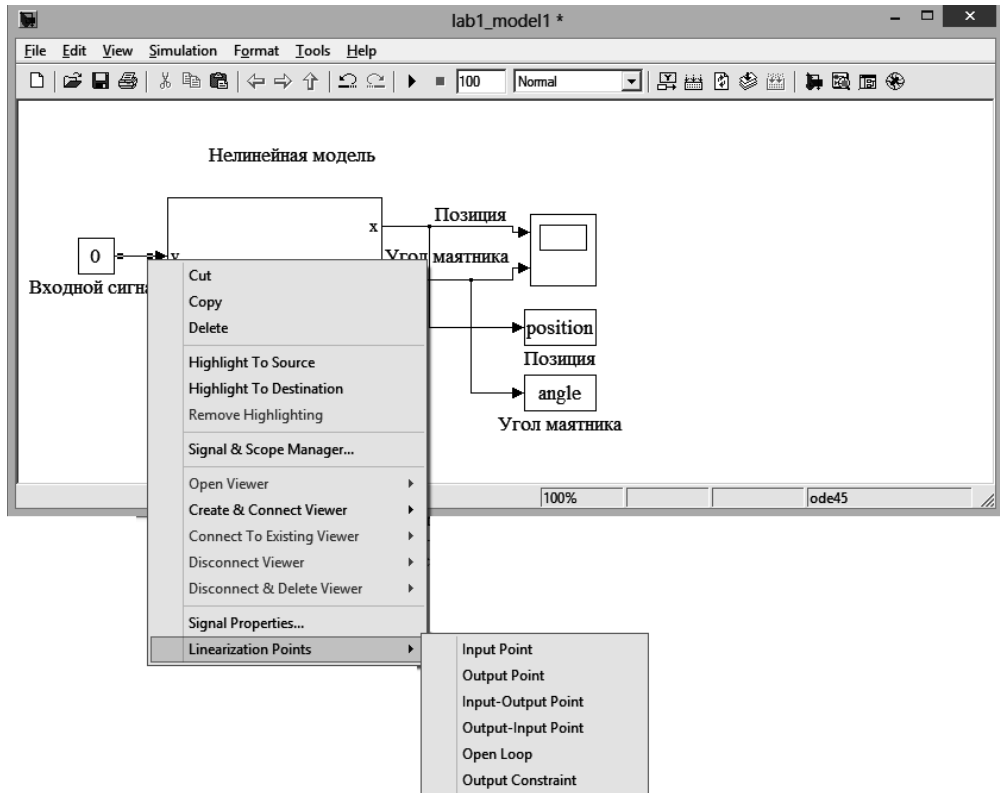


Рис. 1.11. Установка точки Input point

Чтобы провести линеаризацию для верхнего вертикального положения маятника, откройте m-файл `lab1_model1_params.m`, установите начальное угловое положение маятника равным нулю и сохраните m-файл.

Перейдите в модель и выберите в меню `Tools\Control Design` пункт `Linear Analysis`. Запустится `Control and Estimation Tools Manager` (рис. 1.12), в котором будет создан проект с именем «Project — lab1_model1».

Находясь в `Control and Estimation Tools Manager`, выберите `Plot linear analysis result in a step response plot` и нажмите кнопку `Linearize Model`. Отобразится график весовой функции по S и φ . Аналогично можно получить графики для ЛАЧХ и ЛФЧХ (`Bode response plot`), переходной функции (`impulse response plot`), АФХ (`Nyquist plot`), выбрав в выпадающем списке вместо `step`

response plot: Bode response plot для вывода графиков ЛАЧХ и ЛФЧХ, *impulse response plot* для переходной функции, *Nyquist plot* для АФХ.

Сравните полученные графики с соответствующими графиками, сохраненными в п. 1.6.2. Сохраните графики для отчета.

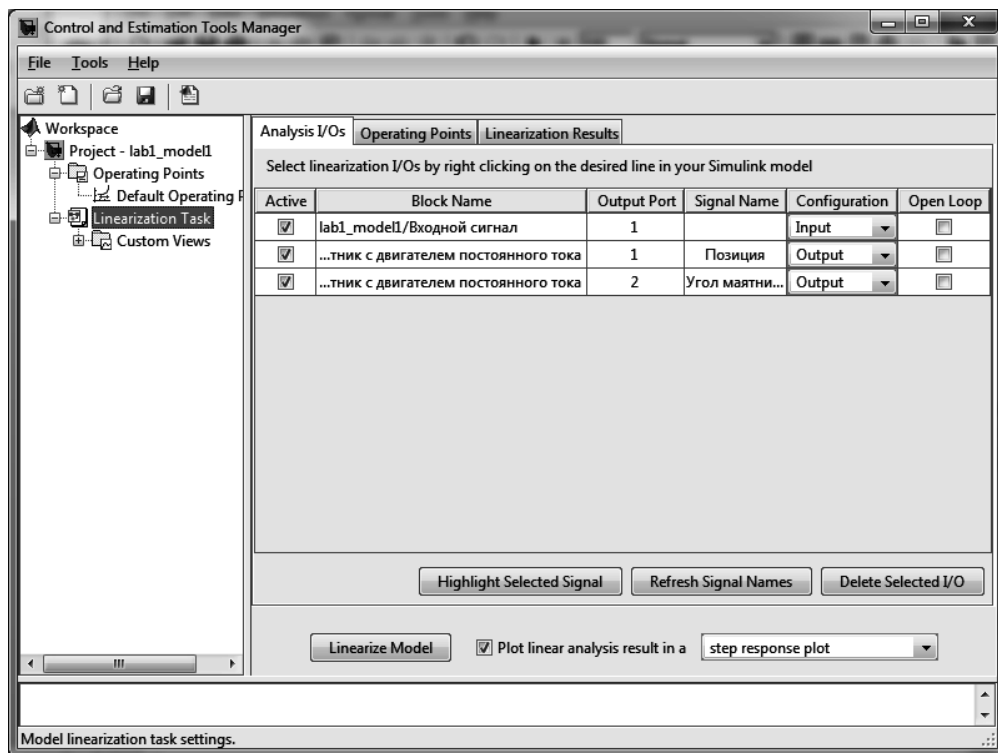


Рис. 1.12. Control and Estimation Tools Manager

Находясь в Control and Estimation Tools Manager, перейдите на вкладку model. В нижней части окна в меню Display linear model as выберете State Space для отображения матриц **A**, **B**, **C**, **D**, Transfer function для отображения передаточной функции (рис. 1.13). Сравните матрицы **A**, **B**, **C**, **D** и передаточную функцию с соответствующими матрицами и передаточной функцией, полученной в п. 1.6.2.

Скопируйте полученную ss-модель в рабочую область. Для этого нажмите кнопку Export to Workspace и задайте в открывшемся окне имя линеаризованной модели. В командном окне для вывода собственных чисел матрицы динамики **A** выполните команду:

eig (Название ss-модели).

Сравните собственные числа с собственными числами матрицы динамики **A**, полученными в п. 1.6.2.

Сохраните матрицы **A**, **B**, **C**, **D** передаточные функции и собственные числа матрицы динамики **A** для отчета.

Закройте Control and Estimation Tools Manager, сохранив проект.

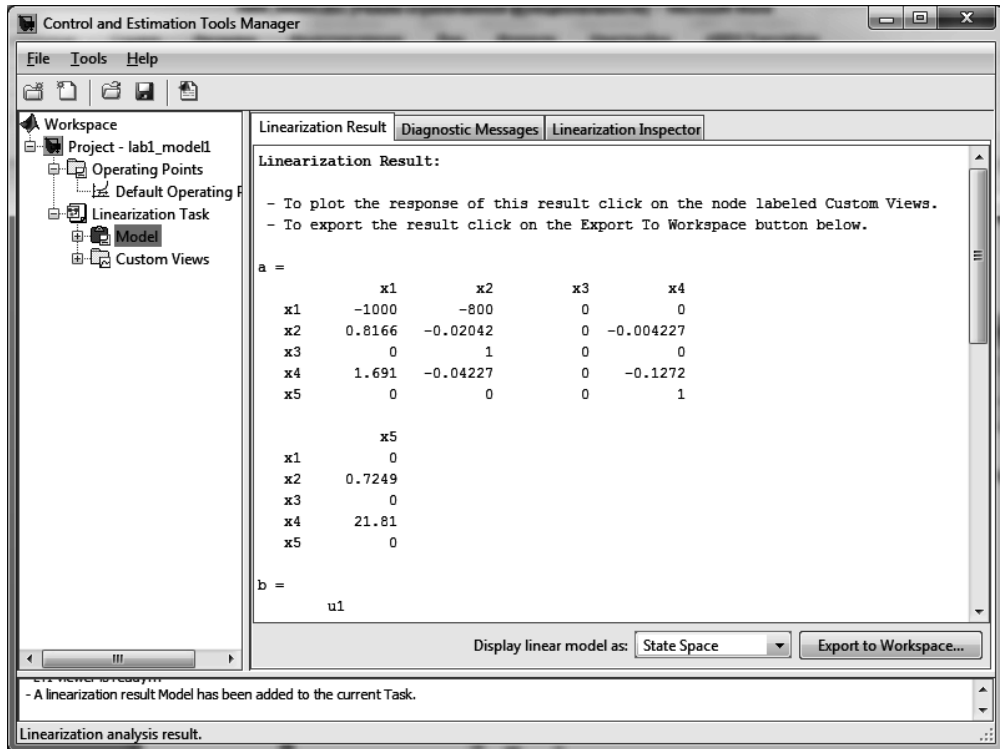


Рис. 1.13. Вкладка Model

Установите в m-файле lab1_model1_params.m начальное угловое положение маятника равным π и сохраните m-файл. С помощью Control and Estimation Tools Manager проведите линеаризацию для нижнего вертикального положения маятника. Получите графики для ЛАЧХ и ЛФЧХ, весовой и переходной функции по S и $\varphi = \pi$, АФХ и сохраните их для отчета. На вкладке Linearization result отобразите передаточную функцию, матрицы **A**, **B**, **C**, **D** вычислите с помощью команды eig собственные числа матрицы динамики **A**. Сравните полученные результаты с соответствующими результатами п. 1.6.2.

1.6.4. Сравнение нелинейной и линеаризованной модели при свободных колебаниях маятника

Откройте файл lab1_model2.mdl, полученный от преподавателя (рис. 1.14). Данный файл предназначен для сравнения нелинейной и линеаризованной

моделей при различном начальном угловом положении маятника. В блок «Объект управления» требуется перенести нелинейную модель объекта, созданную в файле `lab1_model1.mdl`.



Рис. 1.14. Сравнение нелинейной и линеаризованной модели

Исходные данные записаны в m-файле `lab1_model2_params.m` (приведен в прил. 1). Укажите в этом m-файле элементы матрицы динамики A для нижнего вертикального положения маятника ($\varphi = \pi$), полученные в результате предварительной подготовки.

Установите параметр `dphi = 5` (параметр `dphi` указывает отклонение маятника от нижнего вертикального положения в нулевой момент времени для нелинейной модели в градусах) в m-файле `lab1_model2_params.m`. Сохранив m-файл, запустите файл `lab1_model2.mdl`. С помощью m-файла `lab1_plot.m` (приведен в прил. 1) постройте и сохраните графики позиции тележки и углового положения маятника для отчета.

Устанавливая параметр `dphi` в m-файле `lab1_model2_params.m` равным 30 и 60 и сохраняя m-файл, повторите запуск файла `lab1_model2.mdl` и убедитесь в различиях между графиками углового положения маятника для нелинейной и линеаризованной моделей. Определите (рассчитайте и проверьте экспериментально) период колебаний маятника в линеаризованной модели.

Запустите файл `lab1_model2.mdl` при параметре `dphi = 60`. Требуется исследовать изменения периода колебаний маятника в нелинейной модели. Для этого откройте блок «График углового положения маятника» и сделайте набор измерений периода колебаний маятника для нелинейной модели, как показано на рис. 1.15. Измерения необходимо проводить, пока амплитуда колебаний не уменьшится до $1-2^\circ$, должен получиться набор из 20–25 значений периода колебаний маятника.

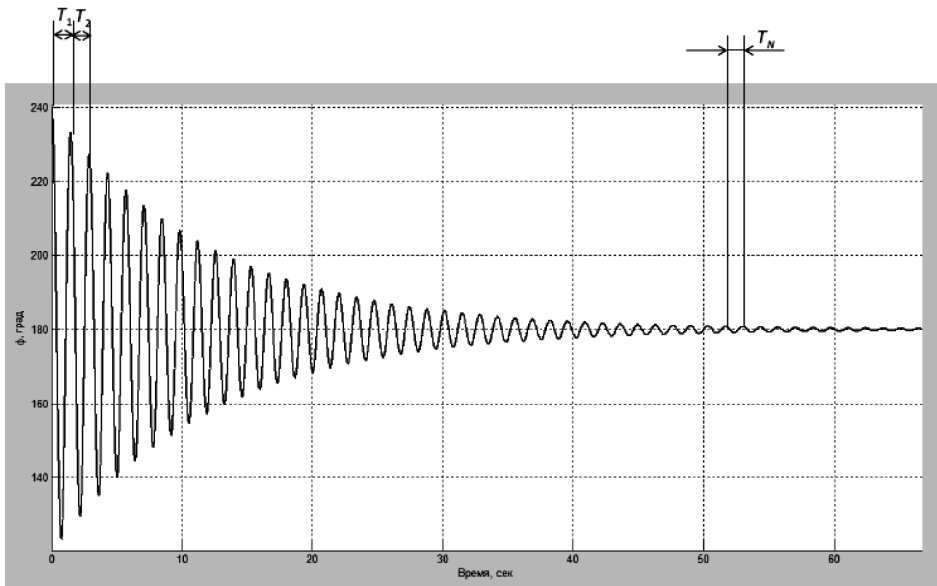


Рис. 1.15. Определение периода и амплитуды колебаний маятника

Измерение рекомендуется проводить с точностью до третьего знака после запятой. Для измерения периода также можно воспользоваться графиком, построенным с помощью m-файла `lab1_plot.m`.

Постройте график значений периода от номера измерения для нелинейной модели. Определите амплитуду, при которой период для нелинейной модели приблизительно равен периоду колебаний маятника для линеаризованной модели. Сохраните график для отчета и сделайте выводы о различиях между нелинейной и линеаризованной моделью.

1.7. Содержание отчета

1. Исходные данные лабораторной работы: название работы, цель и задачи работы и значения параметров, указанных в табл. 1.1, 1.2.
2. Результаты предварительной подготовки.
3. Текст m-файла, используемого при моделировании линеаризованной системы.
4. Результаты выполнения п. 1.6.2: графики весовой и переходной функций, ЛАЧХ, ЛФЧХ и АФХ, нули и полюса передаточной функции, матрицы **A**, **B**, **C**, **D**, собственные числа матрицы динамики **A**.

5. Результаты выполнения п. 1.6.3: графики весовой и переходной функций, ЛАЧХ, ЛФЧХ и АФХ, матрицы **A**, **B**, **C**, **D**, собственные числа матрицы динамики **A** для системы, линеаризованной в верхнем и нижнем вертикальном положении маятника.
6. График зависимости периода от номера измерений для нелинейной модели.
7. Выводы о различиях между нелинейной и линеаризованной моделями.
8. Отчет оформляется на листах формата А4, допускается рукописное, печатное или комбинированное оформление. Листы отчета должны быть скреплены.

Контрольные вопросы

1. Поясните процесс составления математического описания объекта управления.
2. Поясните алгоритм приведения линеаризованных уравнений к векторно-матричным уравнениям в форме Коши.
3. Поясните способы вывода результатов моделирования на графики в среде MATLAB с помощью LTIView.
4. Поясните процесс линеаризации с помощью инструмента Control and Estimation Tools Manager.

Лабораторная работа № 2.

Идентификация параметров объекта управления

2.1. Цели и задачи работы

Цель работы: изучить особенности реализации алгоритмов управления стендом в реальном времени и провести идентификацию параметров объекта управления.

Задачи:

- 1) изучить особенности реализации алгоритмов управления на стенде;
- 2) провести эксперимент по измерению сухого трения тележки;
- 3) сформировать массив данных для проведения идентификации;
- 4) провести предварительную обработку массива данных;
- 5) провести идентификацию параметров объекта управления.

Исходные данные указаны в табл. 2.1, 2.2.

Таблица 2.1

Перечень параметров объекта

Название параметра	Обозначение	Исходное значение	Единица измерения
Масса маятника	m	0,174	кг
Масса тележки	M	1,321	кг
Длина стержня маятника	L	0,38	м
Момент инерции стержня*	J	0,00099	кг·м ²
Коэффициент вязкого трения тележки	b	0,05	Н·с/м
Коэффициент демпфирования маятника*	d	0,005	Н·м·с
Гравитационная постоянная	g	9,81	м/с ²

* Подлежит уточнению в ходе лабораторной работы.

Таблица 2.2

Перечень параметров двигателя

Название параметра	Обозначение	Исходное значение	Единица измерения
Конструктивная постоянная двигателя	$(Cм\Phi)_t = K_t$	0,05	Н·м/А
	$(Cм\Phi)_E = K_b$	0,05	В/ (Рад/с)
Активное сопротивление обмотки ротора	R_a	2,5	Ом
Индуктивность обмотки ротора	L_a	0,0025	Гн
Радиус колеса (шкива)	R_k	0,025	м
Коэффициент вязкого трения на валу двигателя*	$b_{дв}$	0,00568	Н·с·м

* Подлежит уточнению в ходе лабораторной работы

2.2. Особенности работы со стендом в режиме реального времени

2.2.1. Реализация алгоритмов управления с учетом каналов связи с объектом

На ПК в среде Simulink реализуются алгоритмы управления стендом. Также предусмотрена возможность модификации существующих алгоритмов управления и разработки новых. Алгоритмы управления решают две основные задачи: стабилизация перевернутого маятника и управление краном, обеспечивающее перемещение груза в заданное положение с предотвращением его раскачивания.

Рассмотрим внешние интерфейсные блоки Cart Position/Pendulum Angle и Feedback DAC (рис. 2.1), которые используются при реализации алгоритмов управления в режиме реального времени.

Внешний интерфейсный блок Cart Position/Pendulum Angle имеет два выхода: Position — сигнал с датчика положения тележки; Angle — угловое положение маят-

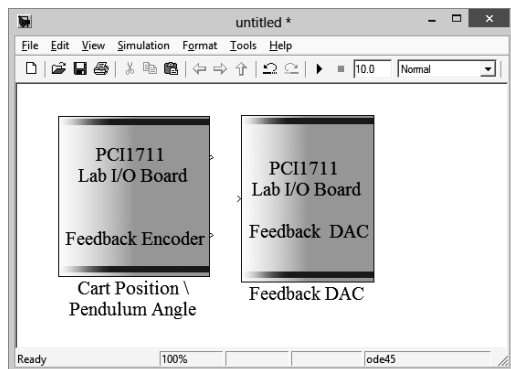


Рис. 2.1. Внешние интерфейсные блоки Cart Position/Pendulum Angle и Feedback DAC

ника. Дважды щелкнув мышкой по блоку, можно изменить параметры этого блока (рис. 2.2). Блок Cart Position\Pendulum Angle имеет 3 параметра: период дискретизации, смещение для 1 и 2 каналов. Первый канал — положение тележки, второй — угловое положение маятника.

Блок Feedback DAC является цифро-аналоговым преобразователем, на который поступает управляющий сигнал. Окно свойств данного блока показано на рис. 2.3. В этом блоке можно указать номер выходного канала и период дискретизации. Интерфейсная плата PCI-1711 имеет 2 аналоговых выхода, выход 1 используется для управления двигателем, выход 2 не используется.

Запомните!

1. При проектировании управления нужно учитывать, что сигнал, поступающий на блок Feedback DAC, должен быть в диапазоне от $-2,5$ В до $2,5$ В. Диапазон $-2,5$ – 0 В соответствует перемещению тележки вправо, диапазон 0 – $2,5$ В — перемещению влево.

2. Значение, которое поступает на аналоговый выход, преобразуется в диапазон от 0 до 5 В внутри блока Feedback DAC.

Рассмотрим преобразование сигнала внутри блока Feedback DAC. Для открытия подсистемы внутри блока необходимо нажать правую кнопку мыши и из появившегося подменю выбрать пункт Look under Mask (Взглянуть под маску). Подсистема блока Feedback DAC показана на рис. 2.4.

Блок Friction Compensation используется для компенсации трения при движении тележки.

После блока компенсации трения сигнал преобразуется в диапазон 0 – 5 В. Для того чтобы сигнал не выходил за этот диапазон в блоке Feedback DAC предусмотрены ограничения Saturation.

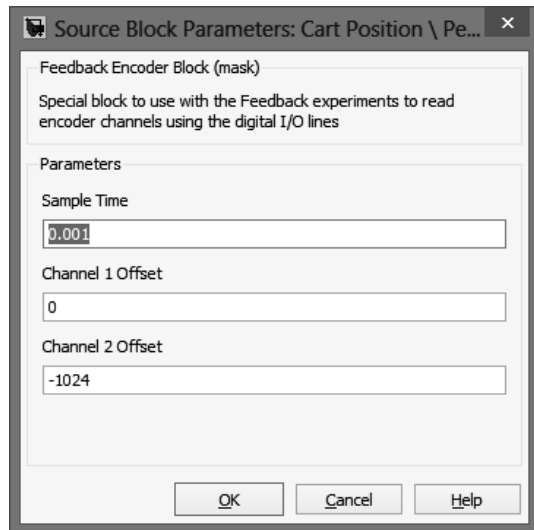


Рис. 2.2. Параметры блока Cart Position\Pendulum Angle

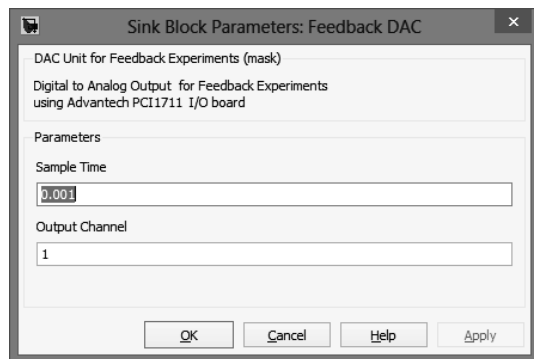


Рис. 2.3. Свойства блока Feedback DAC

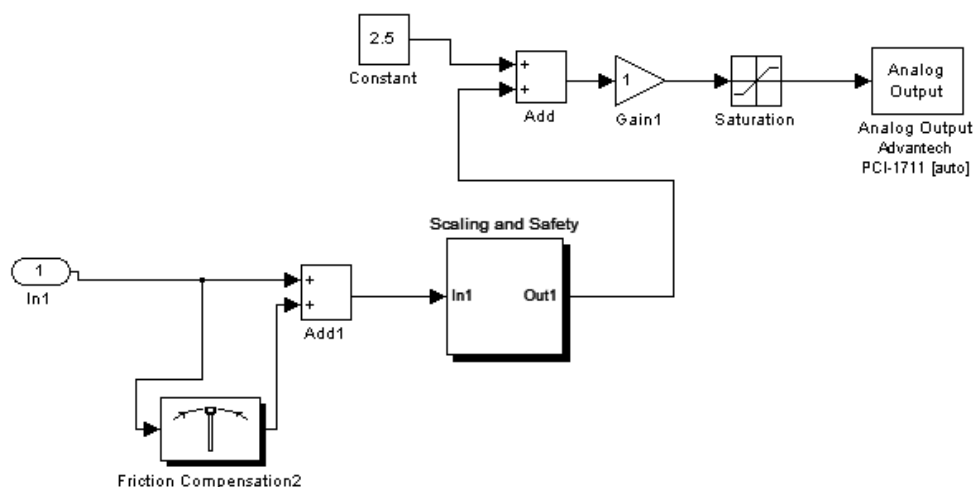



Рис. 2.4. Подсистема блока Feedback DAC

2.2.2. Порядок запуска системы управления в режиме реального времени

Внимательно изучите последовательность запуска системы управления в режиме реального времени. Неправильный порядок запуска может привести к повреждению оборудования.

- Удостоверьтесь, что питание блока управления включено (выключатель слева от кнопки START подсвечен). Зеленая кнопка START не должна быть включена. Если она включена (горит), нажмите кнопку STOP.
- Переместите тележку в нулевую позицию и зафиксируйте маятник в нижнем положении, чтобы он не раскачивался.
- Перед запуском системы в режиме реального времени и после каждой коррекции mdl-файла необходимо скомпилировать алгоритм управления, для этого в меню Tools/Real-Time Workshop выбрать пункт Build model.
- После появления в командном окне сообщения «Successful completion...» соедините программу с объектом, нажав кнопку Connect to Target . Решение еще не будет запущено, но интерфейсная плата PCI-1711 будет правильно инициализирована.
- Убедитесь, что вокруг движущихся частей стенда нет посторонних предметов, мешающих его движению.
- Будьте готовы в любое время нажать кнопку STOP, чтобы не вывести стенд из строя при некорректной работе программы.
- Нажмите зеленую кнопку START на блоке управления маятником.
- Запустите решение, нажав кнопку Run в среде SIMULINK.
- После окончания решения нажмите кнопку STOP.

Соблюдайте указанные меры предосторожности при каждом запуске программы в реальном времени.

2.3. Выполнение работы

2.3.1. Создание личной папки для работы с моделями

На диске D:\ создайте папку с идентификатором: **ЛР2_Фамилия_Группа_Год**. Скопируйте в эту папку m-файлы и mdl-файлы, полученные от преподавателя. Запустите MATLAB и перейдите в созданную папку с файлами. Для этого в окне Current folder введите следующий путь: **D:\Имя созданной папки**.

2.3.2. Измерение сухого трения тележки

Для измерения сухого трения тележки используется файл PendulumFriction.mdl, полученный от преподавателя (рис. 2.5). Откройте файл из личной папки для работы с моделями.

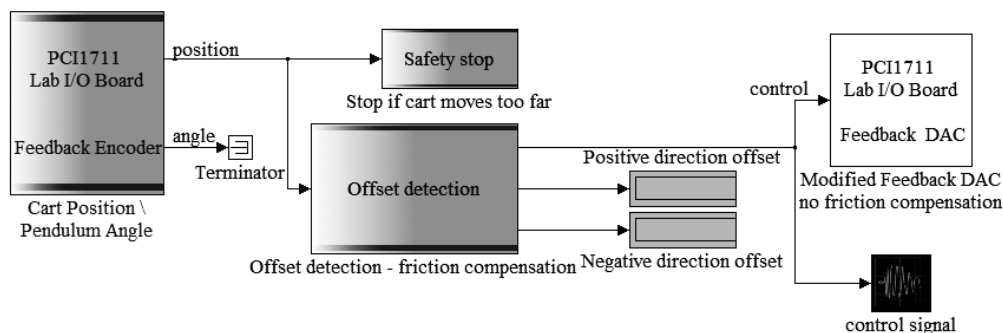


Рис. 2.5. Файл PendulumFriction.mdl

В данном файле измеряется напряжение, которое нужно подать на двигатель для того, чтобы тележка начала движение.

При работе алгоритма управляющее напряжение в период времени от 0 до 10 с увеличивается по линейному закону $u(t) = 0.02t$. Увеличивающееся значение управляющего напряжения отображается в блоке Positive direction offset. Как только тележка начнет перемещение влево, значение напряжения, при котором началось перемещение тележки, фиксируется в блоке Positive direction offset. Если по истечении 10 с от начала моделирования тележка не переместилась влево, то значение в блоке Positive direction offset остается равным 0.02 В (за 10 с напряжение увеличилось до 0.02 В).

В период времени от 10 с до окончания моделирования (25 с) управляющее напряжение уменьшается по линейному закону $u(t) = -0.02t$. Значение управляющего напряжения отображается в блоке Negative direction offset. Как только тележка начнет перемещение вправо, значение напряжения, при котором началось перемещение тележки, фиксируется в блоке Negative direction offset.

После окончания работы модели в блоках Positive direction offset и Negative direction offset будут показаны значения напряжения, при которых тележка начинает движение вправо и влево.

Измеренные значения напряжения требуется учитывать в блоке Friction Compensation (рис. 2.4) для компенсации сухого трения.

Скомпилируйте файл PendulumFriction.mdl. Для этого в меню Tools/Real-Time Workshop выберите пункт Build model.

Чтобы наблюдать продвижение компиляции, необходимо перейти в командное окно MATLAB. При успешном завершении компиляции в командном окне появится сообщение: «**Successful completion of Real-Time Workshop build procedure for model: Название модели**». Если это сообщение не появилось, то компиляция завершилась с ошибками. Необходимо в присутствии преподавателя исправить ошибки и повторить компиляцию.

После успешной компиляции запустите решение в соответствии с последовательностью запуска алгоритма управления, указанной в разделе 2.2.2. После окончания работы алгоритма в блоках Positive direction offset и Negative direction offset появятся значения напряжений, необходимых для перемещения тележки в положительном (U_1) и отрицательном (U_2) направлении. Запишите в отчет значения напряжений U_1 и U_2 .

2.3.3. Формирование массива данных для проведения эксперимента по идентификации

Откройте файл Pendulum_ident.mdl, полученный от преподавателя (рис. 2.6).

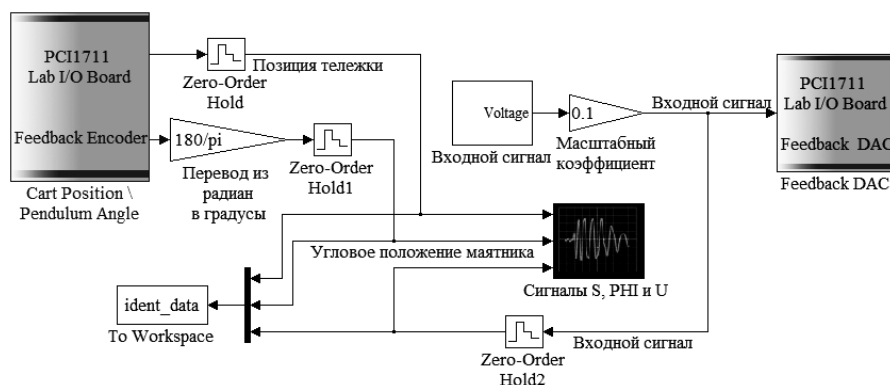


Рис. 2.6. Файл Pendulum_ident.mdl

Данный файл формирует массив выходных данных, которые будут использоваться при проведении эксперимента по идентификации. В блоке «Входной сигнал» (рис. 2.7) формируется сигнал, состоящий из четырех гармонических сигналов:

$$u(t) = \sin(5t + 1) - \sin(15t) + \sin(20t + 0.3) - 0.5\sin(t + 0.5).$$

Дополнительно к входному сигналу можно добавить случайный сигнал с нормальным распределением (блок Random number).

Из сигналов датчиков с помощью блока Zero-Order Hold производится выборка с периодом 0.05 с и результат помещается в рабочую область MATLAB в виде массива `ident_data`. Запись массива `ident_data` в рабочую область организована с помощью блока «To Workspace». В параметрах блока (рис. 2.8) указывается имя массива (Variable name), период дискретизации (Sample time) и формат сохранения данных (Save format). В качестве формата сохранения данных можно выбрать структуру (Structure) или структуру со временем (Structure with time) и массив (Array). Для идентификации требуется сохранения данных в виде массива.

Перед запуском алгоритма в подсистеме блока Feedback DAC требуется указать значения напряжений для компенсации сухого трения тележки. Откройте подсистему внутри блока Feedback DAC, нажав правую кнопку мыши и из появившегося подменю выбрав пункт Look under Mask (взглянуть под маску). В подсистеме откройте блок

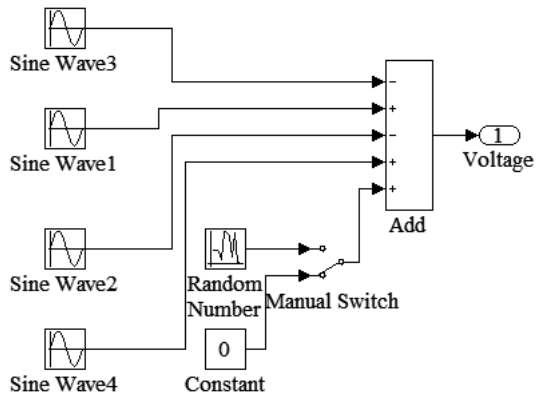


Рис. 2.7. Блок Входной сигнал

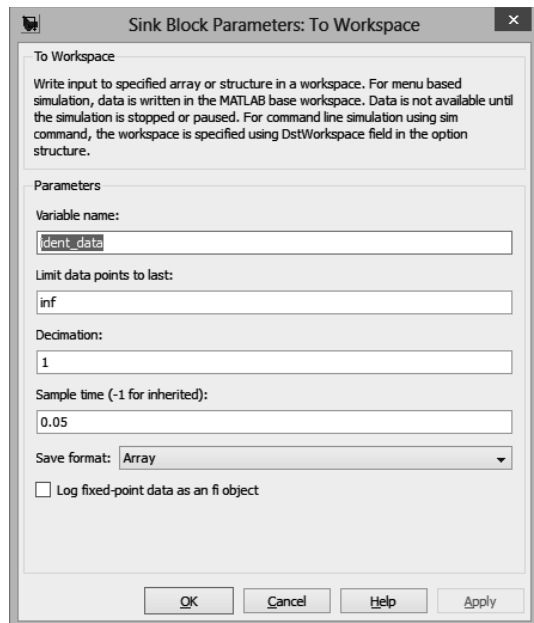


Рис. 2.8. Параметры блока «To Workspace»

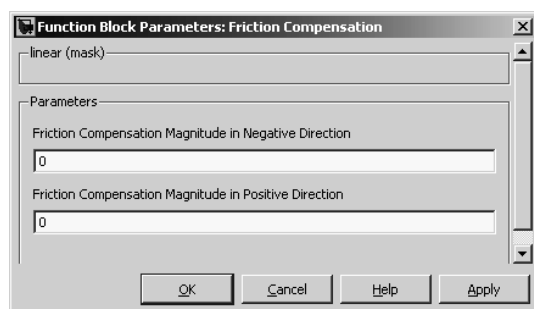


Рис. 2.9. Блок «Friction Compensation»

в положительном направлении (U_1). Нажмите кнопку Apply для принятия внесенных изменений и закройте окно.

Откройте подсистему внутри блока «Friction Compensation» и изучите логику работы блока. Скомпилируйте и запустите файл Pendulum_ident.mdl в соответствии с последовательностью запуска модели, указанной в разделе 2.2.2.

При работе алгоритма возможно появление тренда позиции тележки, при этом наблюдается сильное смещение (тренд) тележки влево или вправо относительно центральной метки на стенде. Под трендом понимается присутствие в процессе систематической составляющей, например, линейно изменяющейся во времени, не обусловленной непосредственно входным сигналом (рис. 2.10). Такой тренд существенно осложняет последующую процедуру идентификации.

Предлагаются следующие способы устранения тренда.

1. Скорректируйте значения напряжений, указанных в блоке «Friction Compensation». Например, при сильном смещении тележки в положительном направлении необходимо уменьшить напряжение U_1 или увеличить напряжение U_2 . Изменять значения напряжений U_1 и U_2 рекомендуется в небольших пределах (0,01–0,02 В). После корректировки напряжений повторите компиляцию и запуск алгоритма. Для просмотра выходных сигналов откройте блок «Сигналы S, fi, U». Если в выходном сигнале позиции тележки наблюдается тренд, повторно скорректируйте значения напряжений в блоке «Friction Compensation», скомпилируйте и запустите алгоритм. Корректировкой значений напряжений U_1 и U_2 добейтесь уменьшения тренда.

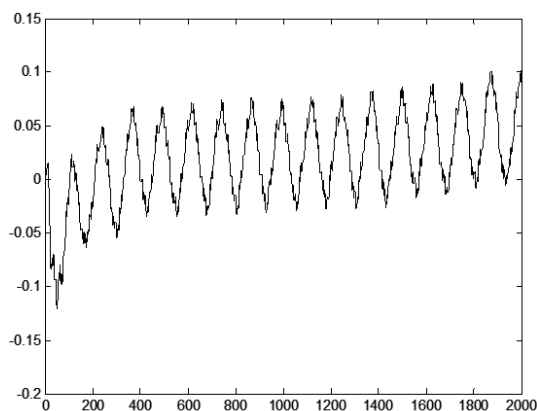


Рис. 2.10. Тренд позиции тележки

«Friction Compensation» (рис. 2.9), в котором укажите значения напряжений U_1 и U_2 для компенсации сухого трения тележки. Обратите внимание, что сначала следует указать абсолютное значение напряжения, необходимого для перемещения тележки в отрицательном направлении ($|U_2|$), затем значение напряжения для перемещения тележки в положи-

2. Измените частоты и фазы гармонических сигналов (блоки Sine wave) в блоке «Входной сигнал». Параметры гармонического сигнала (частоты и фазы) рекомендуется менять в небольших пределах от заданных значений, например можно сформировать следующий входной сигнал:

$$u(t) = \sin(4t + 1.5) - \sin(12t) + \sin(18t + 0.6) - 0.7\sin(0.5t + 1).$$

После изменения параметров входного сигнала скомпилируйте и запустите алгоритм.

Если после указанных в п. 1 и 2 действий тренд не был полностью устранен, то его необходимо устранить при обработке сигнала перед идентификацией. Данный способ устранения тренда будет рассмотрен в разделе 2.3.4.

Запустите программу с параметрами входного сигнала и значениями напряжений U_1 и U_2 , при которых наблюдается минимальный тренд позиции тележки. После окончания работы программы выключите стенд и перейдите в рабочую область (Workspace) MATLAB. Сохраните массив `ident_data`, для этого найдите в столбце Name данный массив, наведите курсор на массив и нажмите правую кнопку мыши. В появившемся контекстном меню выберите пункт `Save as...` и сохраните массив `ident_data` в виде `mat`-файла.

Откройте и запустите файл `lab2_plot1.m` (приведен в прил. 2) для вывода графиков позиции тележки, углового положения маятника и входного напряжения двигателя. Сохраните полученные графики для отчета.

2.3.4. Обработка массива данных перед проведением идентификации

Массив `ident_data` перед процедурой идентификации требуется обработать следующим образом.

1. Исключить из массива фрагмент, в котором происходит обработка начальных условий. Пунктирными линиями на рис. 2.11 показан пример фрагмента, который следует выделить для последующей обработки.

Для выделения фрагмента данных сделайте двойной щелчок на массиве `ident_data`, массив откроется в редакторе переменных (Variable Editor). Постройте график позиции тележки, для этого в командном окне выполните команду:

```
plot(ident_data(:,1)).
```

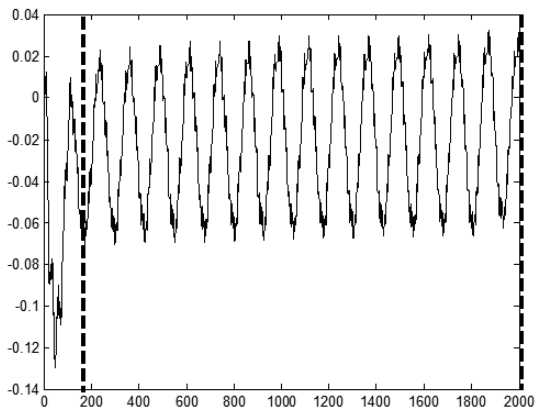


Рис. 2.11. Исключение из массива `S` фрагмента, в котором происходит обработка начальных условий

Определите по графику точки начала и конца необходимого фрагмента данных. Скопируйте выбранный фрагмент данных в новый массив S , выполнив в командном окне следующую команду:

```
S=ident_data(begin_f:end_f,1);
```

где $begin_f$ и end_f — начало и конец выбранного фрагмента данных.

Для идентификации массивы S , Φ и U должны иметь одинаковый размер. Откорректируйте массивы Φ и U аналогично массиву S :

```
 $\Phi$ =ident_data(begin_f:end_f,2);
```

```
U=ident_data(begin_f:end_f,3);
```

2. Провести обработку массива S с целью устранения линейного тренда, если тренд не был полностью устранен в разделе 2.3.3. Для устранения тренда выполните в командном окне следующую команду:

```
S = detrend(S,'linear');
```

Сохраните массивы S , Φ и U в виде mat -файлов, для этого перейдите в рабочую область (Workspace) MATLAB, найдите в столбце Name необходимый массив, наведите курсор на массив и нажмите правую кнопку мыши. В появившемся контекстном меню выберите пункт Save as... и сохраните массив в виде mat -файла.

С помощью файла `lab2_plot2.m` (приведен в прил. 2) отобразите массивы S , Φ и U в виде графиков. В m -файле требуется указать длину каждого фрагмента данных (`length_mass`). Сохраните полученные графики для отчета.

2.3.5. Идентификация параметров объекта с помощью инструмента Control and Estimation Toolbox Manager

Откройте файл `lab2_ident_model.mdl`. Данный файл содержит нелинейную модель объекта (рис. 2.12). С помощью данной модели будут идентифицированы параметры объекта. После открытия данного файла в рабочую область будут записаны начальные значения параметров объекта, содержащиеся в m -файле `lab2_params.m` (приведен в прил. 2).

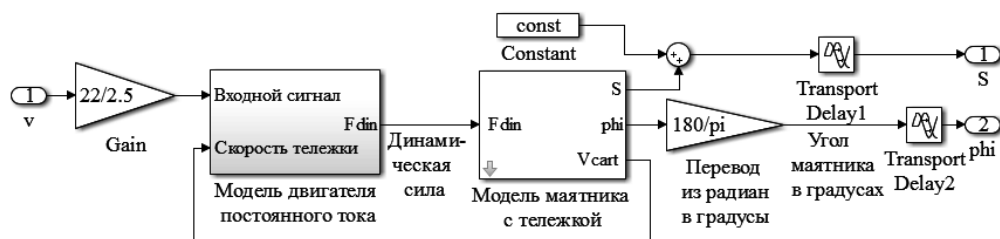


Рис. 2.12. Модель для идентификации параметров объекта

В процессе идентификации выходной сигнал позиции тележки, снятый с объекта, может быть смещен по уровню относительно сигнала позиции

тележки, снятого с модели. Для устранения смещения по уровню в модель добавлена константа, значение которой оценивается в процессе идентификации. Для устранения возможного смещения по фазе после выходных сигналов добавлены блоки транспортного запаздывания (Transport Delay). Время запаздывания также оценивается в процессе идентификации.

Для идентификации требуется массивы S , Φ и U . Загрузите массивы из сохраненных mat-файлов, если они отсутствуют в рабочей области.

Для более точной идентификации необходимо, чтобы начальные условия для положения тележки и углового положения маятника совпадали с первыми элементами массивов S и Φ соответственно. Для этого выполните в командном окне следующие команды:

$s0 = S(1)$ — задание начальных условий для положения тележки.

$\phi0 = \Phi(1) * \pi / 180$ — задание начальных условий (в радианах) для углового положения маятника.

В процессе идентификации будет происходить определение (вычисление) значений параметров нелинейной модели по результатам сравнения измеренных выходов объекта (массивы S , Φ) с процессами на выходе модели.

Через меню Tools/Parameter Estimation запустите средство Simulink Design Optimization, входящие в состав инструмента Control and Estimation Tools Manager (рис. 2.13).

В левой части окна находится дерево проекта, содержащее: задачу оценки (Estimation task), текущие входные и выходные данные (Transient data), переменные (Variables), оценивание (Estimation), проверку достоверности (Validation).

Перейдя на вкладку Estimation task, заполните заголовок задачи, авторов и т.д.

Далее перейдите на вкладку Transient data, нажмите кнопку New. Внутри вкладки Transient data в дереве проекта появится объект New Data, выберите этот объект. Во вкладке Input data в поле Data введите U (входной сигнал), в поле Time/Ts введите период дискретизации 0.05 с (рис. 2.14). Перейдите на вкладку Output data и в поле Data введите S (выходной сигнал), в следующее поле Data введите Φ , в полях Time/Ts введите период дискретизации 0.05 с.

Перейдите на вкладку Variables и на вкладке Estimate Parameters нажмите кнопку Add. Появится окно Select Parameters (рис. 2.15), в котором необходимо выбрать переменные, которые будут идентифицированы.

Выберите для идентификации параметры J , d , $b_{\text{дв}}$, а также вспомогательные параметры $const$ (добавление постоянной составляющей к сигналу позиции тележки), $delay1$, $delay2$ (временные задержки для устранения смещения по фазе между сигналами, снятыми с объекта и выходными сигналами модели). Для этого, нажав клавишу Ctrl, выделите необходимые переменные и нажмите кнопку Apply, выделенные переменные исчезнут из списка. Закрой-

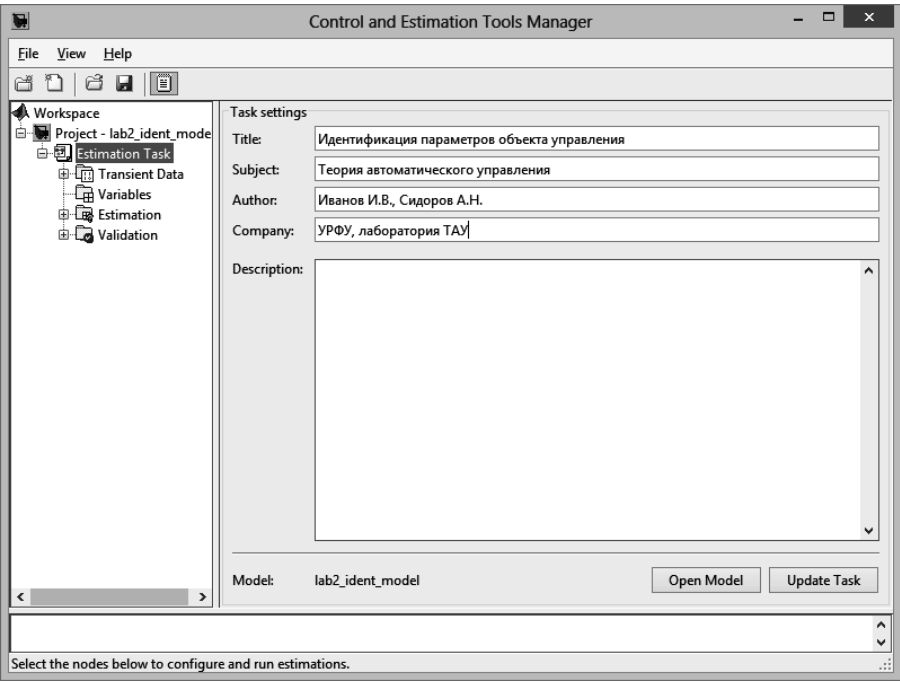


Рис. 2.13. Вкладка Estimation task

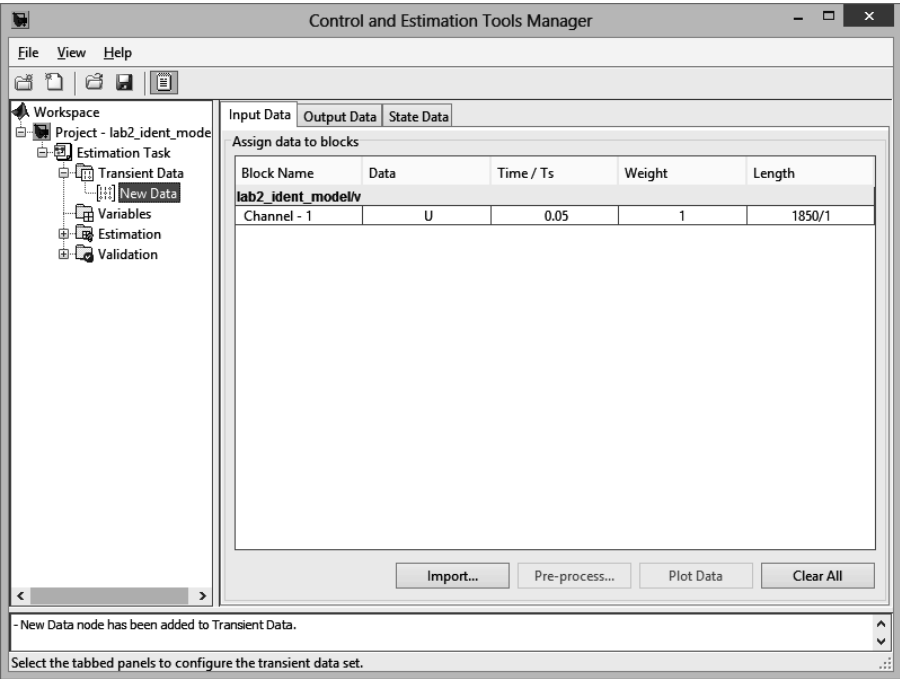


Рис. 2.14. Вкладка Transient data

те окно Select Parameters, нажав кнопку OK. На вкладке Default Settings необходимо задать максимальное и минимальное значение каждого параметра. Задайте максимальные и минимальные значения идентифицируемых параметров согласно табл. 2.3.

В дереве проекта перейдите на вкладку Estimation, нажмите кнопку New, появится новый объект New Estimation. Перейдите в New Estimation и на вкладке Data Set выберите данные для оценивания (поставив галочку в столбце Selected), перейдите на вкладку Parameters и выберите оцениваемые параметры, далее перейдите на вкладку Estimation (рис. 2.16).

Отметьте галочкой опцию Show progress views для графического

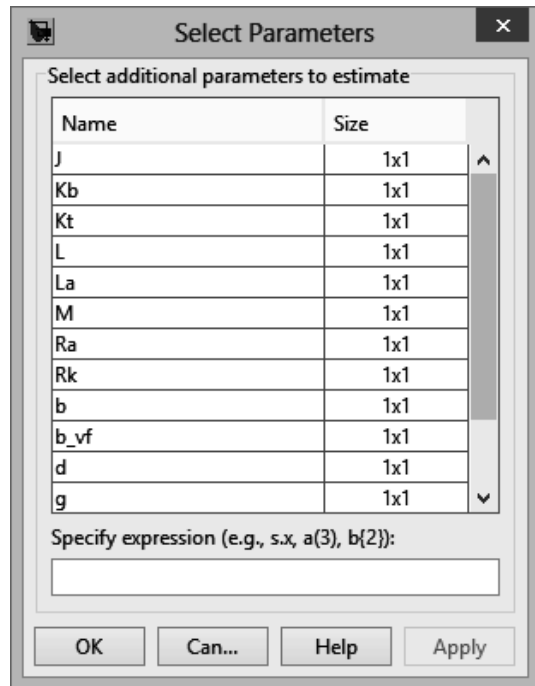


Рис. 2.15. Вкладка Select Parameters

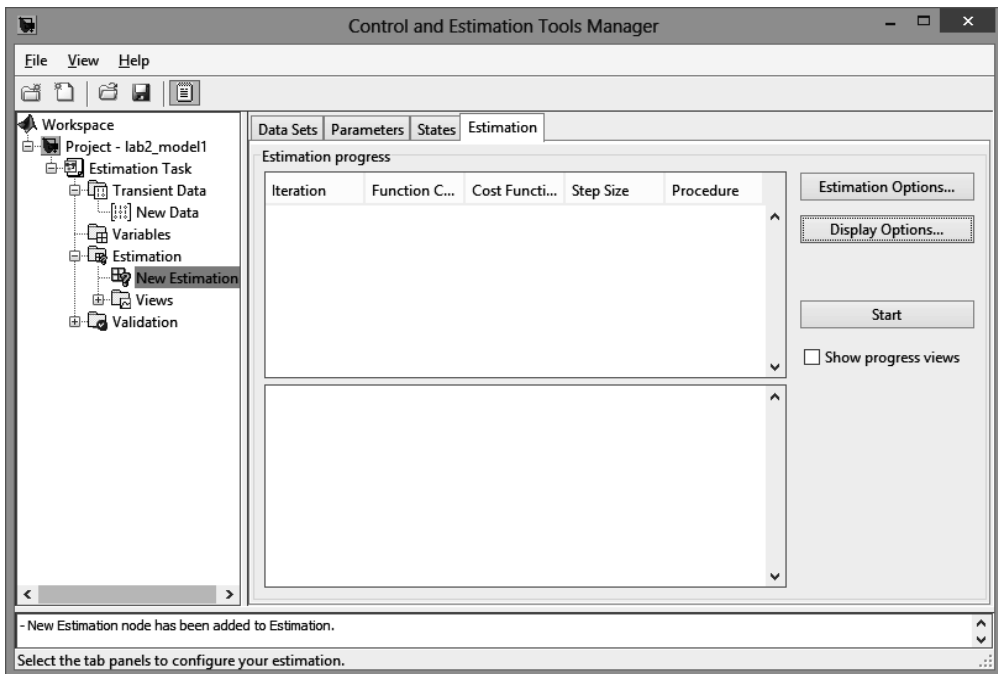


Рис. 2.16. Вкладка Estimation

отображения процесса идентификации параметров объекта и запустите оценивание, нажав кнопку Start. Процесс оценивания (идентификации) может продолжаться длительное время. При этом на вкладке Estimation можно наблюдать за процедурой идентификации. В таблице Estimation progress после каждой итерации отображается значение функционала соответствия модели объекту (Cost Function) и размер шага (Step Size), которые уменьшаются с увеличением количества итераций. В открывшемся окне «Default view — Plot 1 (Parameter trajectory)» можно наблюдать за изменением параметров объекта в процессе идентификации. В окне «Default view — Plot 2 (Measured and simulated)» отображаются выходные сигналы, снятые с объекта (массивы S и PHI), и выходные сигналы нелинейной модели.

Таблица 2.3

Диапазон изменения идентифицируемых параметров

Параметр	Обозначение в MATLAB	Максимальное значение	Минимальное значение
J	J	0.01	0.00001
d	d	1	0.0001
$b_{\text{дв}}$	b_vf	0.1	0.0001
const	const	−1	1
delay1	delay1	0	20
delay2	delay2	0	20

После окончания процесса идентификации запишите в отчет идентифицированные значения параметров, значение функционала соответствия модели объекту (Cost Function) и сохраните графики изменения параметров в процессе оценивания (окно «Default view — Plot 1 (Parameter trajectory)»), выходные сигналы, снятые с объекта (массивы S и PHI), и выходные сигналы нелинейной модели (окно «Default view — Plot 2 (Measured and simulated)»). Сравните идентифицированные параметры с параметрами, представленными в табл. 2.1, 2.2. Продемонстрируйте результаты процесса идентификации преподавателю.

Вычислите общий коэффициент трения b_{Σ} по следующей формуле:

$$b_{\Sigma} = b + \frac{b_{\text{дв_идент}}}{R_{\kappa}^2}, \quad (2.1)$$

где $b_{\text{дв_идент}}$ — идентифицированное значение параметра $b_{\text{дв}}$.

Добавьте к идентифицируемым параметрам коэффициент вязкого трения тележки b . укажите минимальное значение параметра 0,0001, максимальное — 20. Повторите процесс оценивания. После окончания процесса оценивания запишите в отчет идентифицированные параметры и значение функционала

соответствия модели объекту (Cost Function) и сравните их с результатами, полученными в предыдущем запуске процесса оценивания. Вычислите общий коэффициент трения b_{Σ} , подставив в формулу (2.1) идентифицированные значения параметров b и $b_{\text{дв}}$. Сравните полученный коэффициент b_{Σ} с вычисленным ранее и сделайте соответствующие выводы.

Процедура идентификации параметров объекта — сложный, итеративный процесс, алгоритм оценивания может достичь локального минимума при достаточно высоком значении функционала соответствия модели объекту. Если в предыдущих запусках процедуры оценивания было получено значение функционала соответствия модели объекту больше 10000 или амплитуды колебаний сигналов позиции тележки и углового положения маятника сильно различаются, необходимо добиться уменьшения значения функционала. Для этого рекомендуется изменить начальные значения параметров J , d , $b_{\text{дв}}$ для выхода из локального минимума и повторить процесс оценивания. Изменять значения идентифицируемых параметров нужно понимая их физический смысл и влияние на процессы. Уменьшение коэффициента вязкого трения на валу двигателя $b_{\text{дв}}$ приведет к увеличению амплитуды колебаний тележки и маятника. Увеличение коэффициента демпфирования маятника или момента инерции уменьшит амплитуду колебаний маятника.

Изменяя начальные значения параметров J , d , $b_{\text{дв}}$ и повторяя процедуру идентификации, добейтесь уменьшения значения функционала соответствия модели объекту. Для наилучшего варианта запишите в отчет идентифицированные значения параметров, значение функционала соответствия модели объекту (Cost Function) сохраните графики изменения параметров в процессе оценивания (окно «Default view — Plot 1 (Parameter trajectory)»), выходные сигналы, снятые с объекта (массивы S и PNI), и выходные сигналы нелинейной модели (окно «Default view — Plot 2 (Measured and simulated)»). Сравните идентифицированные параметры с параметрами, представленными в табл. 2.1, 2.2. Продемонстрируйте результаты процесса идентификации преподавателю.

2.4. Содержание отчета

1. Исходные данные лабораторной работы: название работы, цель работы и значения параметров, указанных в табл. 2.1, 2.2.
2. Значения напряжений U_1 и U_2 , необходимых для перемещения тележки в положительном и отрицательном направлении, полученные в эксперименте по измерению сухого трения.
3. Графики позиции тележки, углового положения маятника и входного напряжения двигателя, полученные в разделе 2.3.3.

4. Графики позиции тележки, углового положения маятника и входного напряжения двигателя после обработки массивов S , Φ и U , полученные в разделе 2.3.4.
5. Описание процесса идентификации с помощью средства Simulink Design Optimization, результаты оценки параметров объекта.
6. Отчет оформляется на листах формата А4, допускается рукописное, печатное или комбинированное оформление. Листы отчета должны быть скреплены.

Контрольные вопросы

1. Поясните состав лабораторного стенда и взаимодействие его основных частей.
2. Поясните особенности реализации алгоритмов управления на данном стенде, назначение внешних интерфейсных блоков.
3. Укажите порядок запуска системы управления в режиме реального времени.
4. Поясните ход эксперимента по измерению сухого трения тележки.
5. Поясните процесс формирования и обработки массива данных для проведения идентификации.
6. Поясните назначения и принцип работы блока «Friction Compensation».
7. Поясните процесс идентификации с помощью средства Simulink Design Optimization.

Лабораторная работа № 3.

Модальный синтез управления в режиме крана

3.1. Цели и задачи работы

Цель: провести модальный синтез управления стендом в режиме крана.

Задачи:

- 1) изучить состав и принцип работы модального регулятора применительно к рассматриваемому объекту;
- 2) составить линейную модель объекта четвертого порядка;
- 3) путем моделирования исследовать работу модального регулятора при различных собственных числах матрицы динамики замкнутой системы и наблюдателя;
- 4) выбрать параметры модального регулятора на модели объекта в режиме крана;
- 5) проверить работу модального регулятора на стенде в режиме крана.

Исходные данные указаны в табл. 3.1, 3.2.

Таблица. 3.1

Перечень параметров объекта

Название параметра	Обозначение	Исходное значение	Единица измерения
Масса маятника	m	0,174	кг
Масса тележки	M	1,321	кг
Длина стержня маятника	L	0,38	м
Момент инерции стержня	J	— *	кг·м ²
Коэффициент вязкого трения тележки	b	0,05	Н·с/м
Коэффициент демпфирования маятника	d	—	Н·м·с
Гравитационная постоянная	g	9,81	м/с ²

* Укажите уточненное значение параметра, полученное в лабораторной работе № 2

Таблица 3.2

Перечень параметров двигателя

Название параметра	Обозначение	Исходное значение	Единица измерения
Конструктивная постоянная двигателя	$(C\Phi)_t = K_t$	0,05	Н·м/А
	$(C\Phi)_E = K_b$	0,05	В/(Рад/с)
Активное сопротивление обмотки ротора	R_a	2,5	Ом
Индуктивность обмотки ротора	L_a	0,0025	Гн
Радиус колеса (шкива)	R_k	0,025	м
Коэффициент вязкого трения на валу двигателя	$b_{дв}$	—*	Н·с·м

* Укажите уточненное значение параметра, полученное в лабораторной работе № 2

3.2. Подготовительная часть

Система векторно-матричных уравнений пятого порядка, составленная в подготовительной части лабораторной работы № 1, является жесткой, наряду с моделью соответствующей движению тележки и маятника, участвует очень быстрая мода ($T_a = \frac{L_a}{R_a} = 0,001\text{с}$), которая отражает электромагнитную динамику якоря двигателя. Пренебрежем влиянием индуктивности на динамику двигателя. Тогда из (1.15) с учетом (1.7) получим следующее уравнение:

$$U_a = i_a R_a + \frac{K_b}{R_k} \dot{S},$$

откуда

$$i_a = \frac{U_a}{R_a} - \frac{K_b}{R_a R_k} \dot{S}. \quad (3.1)$$

Подставив в (1.16) уравнение для тока якоря (3.1), получим следующие уравнения для динамической силы $\vec{F}_{\text{дин}}$:

$$F_{\text{дин}} = \left(\frac{K_t}{R_a} U_a - \frac{K_t K_b}{R_a R_k} \dot{S} - b_{\text{дв}} \frac{\dot{S}}{R_k} \right) / R_k = \frac{K_t}{R_a R_k} U_a - \left(\frac{K_t K_b}{R_a R_k^2} + \frac{b_{\text{дв}}}{R_k^2} \right) \dot{S}. \quad (3.2)$$

С учетом уравнений (3.2) и (1.20, раздел 1.2.3), получим линеаризованные уравнения для нижнего вертикального положения маятника (режима крана):

$$\begin{cases} (m+M)\ddot{S} + b\dot{S} + mL\ddot{\phi} = \frac{K_{\partial\partial}}{r_a R_k} U_a - \left(\frac{K_t K_b}{r_a R_k^2} + \frac{b}{R_k^2} \right) \dot{S}; \\ (J + mL^2)\ddot{\phi} + d\dot{\phi} + mLg\varphi + mL\ddot{S} = 0. \end{cases} \quad (3.3)$$

Используя систему уравнений (3.3), составьте векторно-матричную систему уравнений четвертого порядка, в которой исключено влияние электромагнитной динамики якоря двигателя.

Для получения векторно-матричных уравнений из системы (3.3) требуется составить систему уравнений вида:

$$\begin{cases} \dot{x}_1 = a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + a_{14}x_4 + b_1 u_{\text{дв}} \\ \dot{x}_2 = a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + a_{24}x_4 + b_2 u_{\text{дв}} \\ \dot{x}_3 = a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + a_{34}x_4 + b_3 u_{\text{дв}} \\ \dot{x}_4 = a_{41}x_1 + a_{42}x_2 + a_{43}x_3 + a_{44}x_4 + b_4 u_{\text{дв}} \end{cases}, \quad (3.4)$$

где $a_{11} \div a_{44}$ — элементы матрицы **A**,

$b_1 \div b_4$ — элементы матрицы **B**,

$u_{\text{дв}}$ — управляющее напряжение двигателя.

Для этого выполните следующие действия:

1) введите вектор состояния \vec{x} и управление **u**:

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}, \quad \mathbf{u} = u_{\text{дв}},$$

где $x_1 = S$, $x_2 = \dot{S}$, $x_3 = \varphi$, $x_4 = \dot{\varphi}$;

2) замените переменные в системе уравнений (3.3) на координаты вектора состояния. Заметьте, что $\dot{x}_1 = \dot{S}$, $\dot{x}_3 = \dot{\varphi}$, поэтому $\dot{x}_1 = x_2$, $\dot{x}_3 = x_4$. Преобразуйте систему (3.3) в систему вида (3.4). В полученной системе уравнений вида (3.4) в правой части не должно остаться производных от координат вектора состояния.

Получив систему уравнений (3.4), необходимо записать матрицы **A** и **B**. Элементы матриц **A** и **B** должны быть представлены в символьной форме, с использованием обозначений параметров объекта и двигателя, приведенных в табл. 3.1, 3.2.

Выходные координаты системы — S и φ , поэтому матрица **C** будет иметь вид:

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

матрица $\mathbf{D}_{2 \times 1}$ — нулевая.

Результат выполнения подготовительной части сдайте на проверку преподавателю.

3.3. Выполнение работы

3.3.1. Создание личной папки для работы с моделями

На диске D:\ создайте папку с идентификатором: **ЛР3_Фамилия_Группа_Год**. Скопируйте в эту папку m-файлы и mdl-файлы, полученные от преподавателя. Запустите MATLAB и перейдите в созданную папку с файлами. Для этого в окне Current folder введите следующий путь: **D:\Имя созданной папки**.

3.3.2. Нелинейная модель стенда для модального синтеза управления в режиме крана

Откройте файл lab3_modelSUCrane.mdl (рис. 3.1). Данный файл содержит нелинейную модель стенда и регулятор, построенный по результатам модального синтеза. Исходные данные для моделирования содержатся в файле lab3_ModelSUCrane_ini.m (приведен в прил. 3).

Выходные данные записываются в рабочую область MATLAB с помощью блока output_data. Данные в этом блоке записываются в виде структуры со временем (Structure with time) и используются для вывода графиков с помощью файла lab3_plot1.m (приведен в прил. 3).

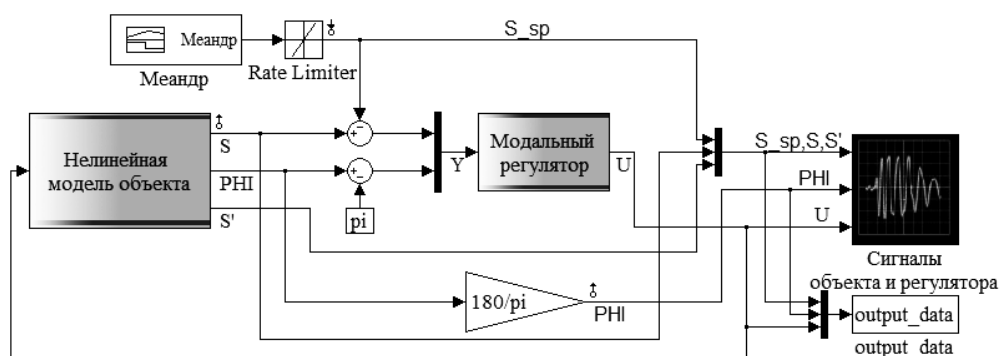


Рис. 3.1. Файл lab3_modelSUCrane.mdl

Нелинейная модель стенда представляет собой полную модель, состоящую из модели двигателя и маятника с тележкой. Нелинейная модель составлена по математическому описанию объекта управления, представленного в разделе 1.2 лабораторной работы № 1.

Блок «Модальный регулятор» (рис. 3.2) состоит из наблюдателя Люенбергера полного порядка, построенного на модели четвертого порядка, и матрицы коэффициентов обратной связи. На вход наблюдателя подается вектор

$$\bar{y} = \{S - S_{sp}; \varphi - \pi\},$$

где S — позиция тележки;

S_{sp} — заданная позиция тележки (уставка);

φ — угловое положения маятника.

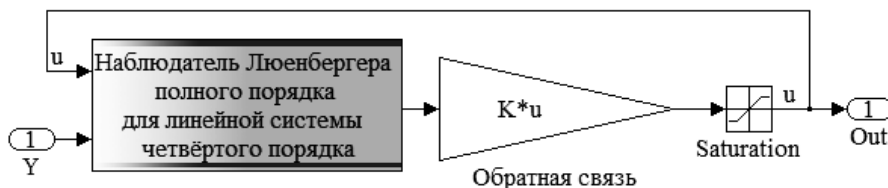


Рис. 3.2. Блок «Модальный регулятор»

Выходом наблюдателя является $\hat{\bar{x}}$ — оценка вектора состояния \bar{x} . Наблюдатель предназначен для восстановления (наблюдения, оценки) вектора \bar{x} по результатам измерения вектора \bar{y} .

Совместно с наблюдателем для управления системой используется обратная связь по состоянию, обеспечивающая желаемое расположение собственных чисел замкнутой системы.

Уставка формируется в виде биполярного меандра с периодом 20 с и амплитудой 0,3 В. Меандр начинается формироваться через 10 с после начала моделирования. Каждые 10 с. амплитуда меандра меняется скачком. **Объект управления не предназначен для отработки скачкообразных воздействий.** Необходимо ограничивать скорость изменения сигнала. Для ограничения скорости входного сигнала используется блок Rate Limiter. Параметры блока имеют следующие значения: Rising slow rate = 0.3, Falling slow rate = -0.3.

Требуется изучить работу блока Rate Limiter. Для этого создайте новый mdl-файл, через меню File/New/Model. Скопируйте в новый mdl-файл блоки Меандр и Rate Limiter. Откройте Simulink Library Browser и добавьте в модель после блока Rate Limiter блок Scope, расположенный в разделе Simulink/Sinks. Установите время моделирования 100 с. Устанавливая различные значения параметров Rising slow rate, Falling slow rate в блоке Rate Limiter, изучите изменения выходного сигнала в блоке Scope. Сделайте выводы о влиянии параметров Rising slow rate, Falling slow rate на скорость изменения выходного сигнала.

3.3.3. Выбор желаемых собственных чисел матриц динамики замкнутой системы и наблюдателя

При выборе желаемых собственных чисел матрицы динамики замкнутой системы следует руководствоваться следующими замечаниями.

1. Желаемые собственные числа матрицы динамики замкнутой системы подбираются так, чтобы элементарные составляющие движений, обусловленные этими собственными числами, успокаивались в несколько раз быстрее, чем результирующие переходные характеристики.
2. Значения показателей качества замкнутой системы зависят от расположения на комплексной плоскости полюсов, ближайших к мнимой оси. При уменьшении соотношения мнимой и действительной части собственных чисел уменьшается перерегулирование, а при увеличении абсолютной величины отрицательной действительной части уменьшается время регулирования.
3. Необходимо обеспечить расположение всех собственных чисел слева от мнимой оси (система должна быть устойчива в замкнутом состоянии).
4. Ближайшей к мнимой оси должна быть пара комплексно сопряженных собственных чисел, не считая собственных чисел, компенсирующих нули систем. Все остальные собственные числа должны находиться значительно левее, чтобы не оказывать влияния на характер переходной функции (соответствующие им моды будут обладать меньшим временем переходного процесса и вносить меньший вклад в выходной сигнал).
5. Следует помнить, что чрезмерный сдвиг собственных чисел влево может привести к значительному увеличению коэффициентов обратных связей, росту пиковых значений управляющего воздействия и появлению нелинейных эффектов в реальных системах, поскольку в них всегда существует ограничение управляющего сигнала.
6. Качество процессов регулирования в основном определяют собственные числа, наиболее близкие к мнимой оси. Целесообразно наиболее близкими к мнимой оси выбирать пару комплексно-сопряженных чисел с вещественной частью — σ и мнимой частью $\pm j\omega$. Тогда время регулирования будет определяться выражением

$$t_{\text{рег}} \approx \frac{3}{\sigma}.$$

Для того чтобы задать первоначальное значение ω , вспомним характеристики типового колебательного звена

$$W_{\text{экр}}(p) = \frac{1}{T_{\text{экр}}^2 p^2 + 2\xi T_{\text{экр}} p + 1}.$$

Полюсы его передаточной функции

$$p_{1,2} = -\frac{\xi}{T_{\text{экр}}} \pm j \frac{\sqrt{1-\xi^2}}{T_{\text{экр}}} = -\sigma \pm j\omega,$$

откуда

$$T_{\text{экв}} = \frac{\xi}{\sigma}; \quad \omega = \frac{\sqrt{1-\xi^2}}{\xi} \sigma.$$

Известно, что переходная функция этого звена при $\xi \approx \frac{1}{\sqrt{2}}$ имеет одновременно минимальное перерегулирование и относительно малое время регулирования. Задаваясь этим значением, по предыдущим формулам рассчитаем сначала σ , а затем ω . При этом получим, что $\omega = \sigma$, то есть, мнимая и действительная части полюсов равны между собой.

Откройте файл `lab3_ModelSUCrane_ini.m` и укажите в нем матрицы A , B , C , D для модели четвертого порядка, полученные в подготовительной части, а также уточненные значения параметров J , d , $b_{\text{дв}}$, полученные в лабораторной работе № 2.

Запустите `m`-файл, произойдет расчет собственных чисел матрицы динамики A , передаточной функции системы четвертого порядка и ее нулей и полюсов.

Изучив рассчитанные собственные числа матрицы динамики A , задайте в файле `lab3_ModelSUCrane_ini.m` желаемые собственные числа матрицы динамики замкнутой системы с учетом указанных выше замечаний 1–6. Например:

`P4_SuCart = [-4-8-2+i*2-2-i*2]` — желаемые собственные числа матрицы динамики замкнутой системы.

Для правильного функционирования наблюдателя необходимо, чтобы переходные процессы в наблюдателе затухали значительно быстрее, чем в объекте управления. Для обеспечения заданных динамических характеристик собственные значения матрицы наблюдателя выбираются такими, чтобы на комплексной плоскости они находились в несколько раз дальше от мнимой оси, чем полюсы передаточной функции объекта.

Укажите в файле `lab3_ModelSUCrane_ini.m` следующие команды:

`Kn = 2` — коэффициент пропорциональности между собственными числами матриц динамики замкнутой системы и наблюдателя.

`P4nab = Kn*P4_SuCart` — собственные числа матрицы наблюдателя.

Организуйте расчет матрицы обратной связи замкнутой системы и матрицы коэффициентов обратных связей наблюдателя. Для этого в файле `lab3_ModelSUCrane_ini.m` введите следующие команды:

`L4_SuCartPlace = place(A4_plant_Cart, B4_plant_Cart, P4_SuCart);` — расчет матрицы обратной связи.

`K4_Obs = place(A4_plant_Cart', C4_plant_Cart', P4nab)'` — расчет матрицы коэффициентов обратных связей наблюдателя, где: `A4_plant_Cart`, `B4_plant_Cart`, `C4_plant_Cart` — матрицы для модели четвертого порядка, полученные в подготовительной части, `P4_SuCart` — собственные числа матрицы динамики замкнутой системы, `P4nab` — собственные числа матрицы наблюдателя.

Сохраните откорректированный файл `lab3_ModelSUCrane_ini.m`.

Требуется исследовать работу модального регулятора при различных собственных числах матриц динамики замкнутой системы и наблюдателя. Для этого выполните следующие действия:

- 1) запустите модель `lab3_ModelSUCrane.mdl`. Откройте блок «Выходные сигналы объекта и регулятора». Определите следующие основные показатели качества замкнутой системы: перерегулирование и время регулирования позиции тележки, максимальную амплитуду углового положения маятника. Результаты занесите в табл. 3.3;
- 2) в `m`-файле `lab3_ModelSUCrane_ini.m` задайте новые собственные числа матрицы динамики A с целью улучшить показатели качества замкнутой системы. Сохраните `m`-файл.

Повторите п. 1 и 2, количество различных вариантов выбора собственных чисел — 7–10. Требуется подобрать такие собственные числа матрицы динамики, при которых показатели качества замкнутой системы минимальны.

Таблица 3.3

Выбор собственных чисел матрицы динамики замкнутой системы

№ варианта	Собственные числа матрицы динамики замкнутой системы	Перерегулирование позиции тележки	Время регулирования позиции тележки	Максимальная амплитуда углового положения маятника
1...10				

Запустите модель `lab3_modelSUCrane` с наилучшим вариантом собственных чисел матрицы динамики замкнутой системы.

Запустите файл `lab3_plot1.m` для отображения на комплексной плоскости исходных и желаемых собственных чисел матриц динамики замкнутой системы и наблюдателя, построения графиков заданного и фактического положения тележки, скорости тележки, угла маятника. Скопируйте в отчет полученные графики с помощью команды меню `Edit\Copy Figure`.

Изменяя в `m`-файле `lab3_ModelSUCrane_ini.m` коэффициент пропорциональности K_p между собственными числами матриц динамики замкнутой системы и наблюдателя, изучите влияние собственных чисел наблюдателя на показатели качества замкнутой системы. Заполните табл. 3.4.

Запустите модель `lab3_modelSUCrane` с наилучшим вариантом собственных чисел матриц динамики замкнутой системы и наблюдателя.

Запустите файл `lab3_plot1.m` для отображения на комплексной плоскости исходных и желаемых собственных чисел матриц динамики замкнутой системы и наблюдателя, построения графиков заданного и фактического положения тележки, скорости тележки, угла маятника. Скопируйте в отчет полученные графики с помощью команды меню `Edit\Copy Figure`.


Таблица 3.4

Выбор собственных чисел матрицы динамики наблюдателя

Коэффициент пропорциональности Kp	Собственные числа матрицы динамики наблюдателя	Перерегулирование позиции тележки	Время регулирования позиции тележки	Максимальная амплитуда углового положения маятника
0,5				
1				
1,5				
2				
3				
4				
5				
10				

3.3.4. Порядок запуска системы управления в режиме реального времени

Внимательно изучите последовательность запуска системы управления в режиме реального времени. Неправильный порядок запуска может привести к повреждению оборудования.

- Удостоверьтесь, что питание блока управления включено (выключатель слева от кнопки START подсвечен). Зеленая кнопка START не должна быть включена. Если она включена (горит), нажмите кнопку STOP
- Переместите тележку в нулевую позицию и зафиксируйте маятник в нижнем положении, чтобы он не раскачивался.
- Перед запуском системы в режиме реального времени и после каждой коррекции mdl-файла необходимо скомпилировать алгоритм управления, для этого в меню Tools\Real-Time Workshop выбрать пункт Build model.
- После появления в командном окне сообщения «Successful completion...» соедините программу с объектом, нажав кнопку Connect to Target . Решение еще не будет запущено, но интерфейсная карта PCI-1711 будет правильно инициализирована.
- Убедитесь, что вокруг движущихся частей стенда нет посторонних предметов, мешающих его движению.
- Будьте готовы в любое время нажать кнопку STOP, чтобы не вывести стенд из строя при некорректной работе программы.
- Нажмите зеленую кнопку START на блоке управления маятником.

- Запустите решение, нажав кнопку Run в среде SIMULINK.
- После окончания решения нажмите кнопку STOP.

Соблюдайте указанные меры предосторожности при каждом запуске программы в реальном времени.

3.3.5. Проверка результатов модального синтеза на стенде

Для проверки результатов модального синтеза на стенде откройте файл lab3_SUCrane.mdl (рис. 3.3). В данном файле для управления стендом в режиме крана используется модальный регулятор, настроенный в разделе 3.3.3. Блок «Согласование с аппаратурой» используется для преобразования выходного сигнала регулятора в диапазоне от -24 В до 24 В в диапазон от $-2,5$ В до $2,5$ В. Исходные данные для работы с файлом lab3_SUCrane.mdl находятся в m-файле lab3_modelSUCrane_ini.m. В данном файле укажите собственные числа матрицы динамики замкнутой системы и наблюдателя, выбранные в разделе 3.3.3, и сохраните файл.

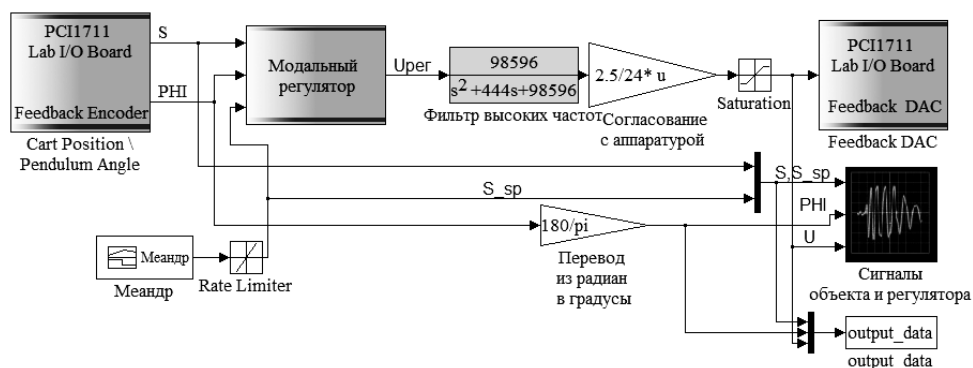


Рис. 3.3. Файл lab3_SUCrane.mdl

Выходные данные записываются в рабочую область MATLAB с помощью блока output_data. Данные в этом блоке записываются в виде структуры со временем (Structure with time) и используются для вывода графиков с помощью файла lab3_plot2.m (приведен в прил. 3).

При работе модального регулятора на стенде в режиме реального времени могут возникнуть высокочастотные колебания управляющего сигнала регулятора. Для обработки выходного сигнала регулятора с целью устранения высокочастотных колебаний используется фильтр высоких частот.

Входной сигнал формируется в блоке «Меандр» и описан в разделе 3.3.2.

Перед запуском стенда внимательно изучите порядок запуска системы управления в режиме реального времени (раздел 3.3.4).

Под руководством преподавателя скомпилируйте файл lab3_SUCrane.mdl и запустите решение. После окончания решения с помощью файла lab3_

plot2.m получите графики заданного и фактического положения тележки, углового положения маятника, управляющего сигнала. Скопируйте в отчет полученные графики с помощью команды меню Edit\Copy Figure.

Сравните графики заданного и фактического положения тележки, углового положения маятника, управляющего сигнала с графиками, полученными в разделе 3.3.3. Сделайте выводы о соответствии результатов моделирования в работе стенда в реальном времени.

3.4. Содержание отчета

1. Исходные данные лабораторной работы: название работы, цель работы, параметры объекта управления, указанные в табл. 3.1, 3.2.
2. Результаты предварительной подготовки.
3. Процесс выбора желаемых собственных чисел матриц динамики замкнутой системы и наблюдателя (табл. 3.3, 3.4). Для наилучшего варианта приведите графики исходных и заданных собственных чисел матриц динамики замкнутой системы и наблюдателя, заданного и фактического положения тележки, скорости тележки, углового положения маятника.
4. Результаты работы модального регулятора на стенде. Графики заданного и фактического положения тележки, углового положения маятника, управляющего сигнала. Выводы о качестве работы замкнутой системы с модальным регулятором и соответствии результатов моделирования работе стенда в реальном времени.
5. Отчет оформляется на листах формата A4, допускается рукописное, печатное или комбинированное оформление. Листы отчета должны быть скреплены.

Контрольные вопросы

1. Поясните состав лабораторного стенда и взаимодействие его основных частей.
2. Укажите порядок запуска системы управления в режиме реального времени.
3. Поясните алгоритм получения векторно-матричного описания линеаризованной системы четвертого порядка.
4. Поясните состав модального регулятора.
5. Поясните процесс выбора желаемых собственных чисел матриц динамики замкнутой системы и наблюдателя.

Лабораторная работа № 4.

Модальный синтез управления в режиме инвертированного маятника

4.1. Цели и задачи работы

Цель: провести модальный синтез управления стендом в режиме инвертированного маятника.

Задачи:

- 1) составить линейную модель объекта четвертого порядка для режима инвертированного маятника;
- 2) путем моделирования исследовать работу модального регулятора для стабилизации инвертированного маятника при различных полюсах замкнутой системы и собственных числах матрицы динамики наблюдателя;
- 3) выбрать параметры модального регулятора на модели объекта в режиме инвертированного маятника;
- 4) проверить работу модального регулятора на стенде в режиме инвертированного маятника.

Исходные данные указаны в табл. 4.1, 4.2.

Таблица. 4.1

Перечень параметров объекта

Название параметра	Обозначение	Исходное значение	Единица измерения
Масса маятника	m	0,174	кг
Масса тележки	M	1,321	кг
Длина стержня маятника	L	0,38	м
Момент инерции стержня	J	—*	кг·м ²
Коэффициент вязкого трения тележки	b	0.05	Н·с/м
Коэффициент демпфирования маятника	d	—*	Н·м·с
Гравитационная постоянная	g	9,81	м/с ²

* Укажите уточненное значение параметра, полученное в лабораторной работе № 2

Таблица 4.2

Перечень параметров двигателя

Название параметра	Обозначение	Исходное значение	Единица измерения
Конструктивная постоянная двигателя	$(\text{См}\Phi)_t = K_t$	0,05	Н·м/А
	$(\text{См}\Phi)_E = K_b$	0,05	В/(Рад/с)
Активное сопротивление обмотки ротора	R_a	2,5	Ом
Индуктивность обмотки ротора	L_a	0,0025	Гн
Радиус колеса (шкива)	R_k	0,025	м
Коэффициент вязкого трения на валу двигателя	$b_{\text{дв}}$	—*	Н·с·м

* Укажите уточненное значение параметра, полученное в лабораторной работе № 2

4.2. Подготовительная часть

Требуется составить векторно-матричную систему уравнений четвертого порядка, линеаризованную для верхнего вертикального положения маятника, исключив влияние электромагнитной динамики якоря двигателя.

Подставим в систему уравнений (1.19) выражение для динамической силы (3.2). В результате получим линеаризованные уравнения в окрестности верхнего вертикального положения маятника (режим инвертированного маятника):

$$\begin{cases} (m + M)\ddot{S} + b\dot{S} - mL\ddot{\varphi} = \frac{K_t}{R_a R_k} U_a - \left(\frac{K_t K_b}{R_a R_k^2} + \frac{b_{\text{дв}}}{R_k^2} \right) \dot{S}; \\ (J + mL^2)\ddot{\varphi} + d\dot{\varphi} - mLg\varphi - mL\dot{S} = 0. \end{cases} \quad (4.1)$$

Для получения векторно-матричных уравнений из системы (4.1) требуется ввести вектор состояния \vec{x} и управление \mathbf{u} :

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} S \\ \dot{S} \\ \varphi \\ \dot{\varphi} \end{pmatrix}, \quad \mathbf{u} = u_{\text{дв}}$$

и составить систему уравнений вида:

$$\begin{cases} \dot{x}_1 = a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + a_{14}x_4 + b_1u_{\text{дв}} \\ \dot{x}_2 = a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + a_{24}x_4 + b_2u_{\text{дв}} \\ \dot{x}_3 = a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + a_{34}x_4 + b_3u_{\text{дв}} \\ \dot{x}_4 = a_{41}x_1 + a_{42}x_2 + a_{43}x_3 + a_{44}x_4 + b_4u_{\text{дв}} \end{cases}, \quad (4.2)$$

где $a_{11} \div a_{44}$ — элементы матрицы A ,

$b_1 \div b_4$ — элементы матрицы B ,

$u_{\text{дв}}$ — управляющее напряжение двигателя.

Преобразование системы уравнений (4.1) к (4.2) аналогично преобразованию системы (3.3) к векторно-матричным уравнениям для режима крана в лабораторной работе № 3.

Получив систему уравнений (4.2), необходимо записать матрицы A и B . Элементы этих матриц должны быть представлены в символьной форме, с использованием обозначений параметров объекта и двигателя, приведенных в табл. 4.1, 4.2.

Выходные координаты системы — S и φ , поэтому матрица C имеет вид:

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

матрица $D_{2 \times 1}$ — нулевая.

Результат выполнения подготовительной части сдайте на проверку преподавателю.

4.3. Выполнение работы

4.3.1. Создание личной папки для работы с моделями

На диске D:\ создайте папку с идентификатором: ЛР4_Фамилия_Группа_Год. Скопируйте в эту папку m-файлы и mdl-файлы, полученные от преподавателя. Запустите MATLAB и перейдите в созданную папку с файлами. Для этого в окне Current folder введите следующий путь: D:\Имя созданной папки.

4.3.2. Моделирование работы стенда в режиме инвертированного маятника

Откройте файл lab4_ModelSUPendulum.mdl (рис. 4.1). Данный файл содержит нелинейную модель стенда и модальный регулятор. Исходные данные для моделирования содержатся в файле lab4_ModelSUPendulum_ini.m.

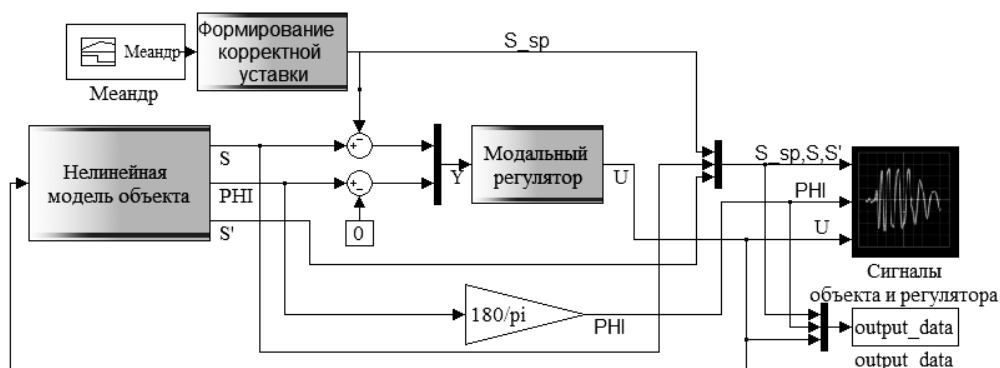


Рис. 4.1. Файл lab4_ModelSUPendulum.mdl

Выходные данные записываются в рабочую область MATLAB с помощью блока output_data. Данные в этом блоке записываются в виде структуры со временем (Structure with time) и используются для вывода графиков с помощью файла lab4_plot1.m.

Нелинейная модель стэнда представляет собой полную модель, состоящую из модели двигателя и маятника с тележкой. Нелинейная модель составлена в соответствии с математическим описанием объекта управления, представленным в п. 1.2 лабораторной работы № 1.

Блок «Модальный регулятор» описан в лабораторной работе № 3.

Уставка формируется в виде биполярного меандра с периодом 20 с и амплитудой 0,3 В. Меандр начинает формироваться через 10 с после начала моделирования. В блоке «Формирование корректной уставки» (рис. 4.2) ограничивается скорость нарастания и сглаживаются фронты сигнала.

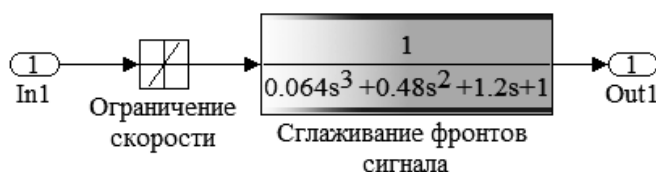


Рис. 4.2. Блок «Формирование корректной уставки»

Откройте файл lab4_ModelSUPendulum_ini.m и запишите в нем символьные имена элементов матриц A, B, C, D для модели четвертого порядка, полученные в подготовительной части, а также уточненные значения параметров J , d , $b_{дв}$, полученные в лабораторной работе № 2.

Запустите m-файл, произойдет расчет передаточной функции системы четвертого порядка, а также ее нулей и полюсов.

Изучив рассчитанные собственные числа матрицы динамики A, задайте в файле lab4_ModelSUCrane_ini.m (приведен в прил. 4) желаемые собствен-

ные числа матрицы динамики замкнутой системы, с учетом замечаний, указанных в разделе 3.3.3 лабораторной работы № 3. Например:

$P4_SuPendulum = [-4-8 \quad -2+i*2 \quad -2-i*2]$ — желаемые собственные числа матрицы динамики замкнутой системы.

Для правильного функционирования наблюдателя необходимо, чтобы переходные процессы в наблюдателе затухали значительно быстрее, чем в объекте управления. Для обеспечения заданных динамических характеристик собственные значения матрицы наблюдателя выбираются такими, чтобы на комплексной плоскости они находились в несколько раз дальше от мнимой оси, чем полюсы передаточной функции объекта.

Укажите в файле `lab4_ModelSUPendulum_ini.m` следующие команды.

$K_n = 2$ — коэффициент пропорциональности между собственными числами матриц динамики замкнутой системы и наблюдателя.

$P4_{nab} = K_n * P4_SuPendulum$ — собственные числа матрицы наблюдателя.

Организируйте расчет матрицы обратной связи замкнутой системы и матрицы коэффициентов обратных связей наблюдателя. Для этого в m-файле `lab4_ModelSUPendulum_ini.m` введите следующие команды:

$L4_SuPendulumPlace = place(A4_Pendulum, B4_Pendulum, P4_SuPendulum);$ — расчет матрицы обратной связи.

$K4_Obs = place(A4_Pendulum, C4_Pendulum', P4_{nab})'$ — расчет матрицы коэффициентов обратных связей наблюдателя, где: $A4_Pendulum$, $B4_Pendulum$, $C4_Pendulum$ — матрицы для модели четвертого порядка, полученные в подготовительной части, $P4_SuPendulum$ — собственные числа матрицы динамики замкнутой системы, $P4_{nab}$ — собственные числа матрицы наблюдателя.

Сохраните откорректированный m-файл `lab4_ModelSUPendulum_ini.m`.

Требуется исследовать работу модального регулятора в режиме инвертированного маятника при различных собственных числах матриц динамики замкнутой системы и наблюдателя. Для этого выполните следующие действия:

- 1) запустите модель `lab4_ModelSUPendulum.mdl`. Откройте блок «Выходные сигналы объекта и регулятора». Определите следующие основные показатели качества замкнутой системы: перерегулирование и время регулирования позиции тележки, максимальную амплитуду углового положения маятника. Результаты занесите в табл. 4.3;
- 2) в m-файле `lab4_ModelSUPendulum_ini.m` задайте новые собственные числа матрицы динамики с целью улучшить показатели качества замкнутой системы. Сохраните m-файл.

Повторите п. 1 и 2, количество различных вариантов выбора собственных чисел — 7–10. Требуется подобрать такие собственные числа матрицы динамики, при которых показатели качества замкнутой системы минимальны.

Запустите модель `lab4_modelSUPendulum` с наилучшим вариантом собственных чисел матрицы динамики замкнутой системы.

Таблица 4.3

Выбор собственных чисел матрицы динамики замкнутой системы

№ варианта	Собственные числа матрицы динамики замкнутой системы	Перерегулирование позиции тележки	Время регулирования позиции тележки	Максимальная амплитуда углового положения маятника
1...10				

Запустите файл `lab4_plot1.m` для отображения на комплексной плоскости исходных и желаемых собственных чисел матриц динамики замкнутой системы и наблюдателя, построения графиков заданного и фактического положения тележки, скорости тележки, угла маятника. Скопируйте в отчет полученные графики с помощью команды меню `Edit\Copy Figure`.

Изменяя коэффициент пропорциональности Kp между собственными числами матриц динамики замкнутой системы и наблюдателя, изучите влияние собственных чисел наблюдателя на показатели качества замкнутой системы. Заполните табл. 4.4.

Таблица 4.4

Выбор собственных чисел матрицы динамики наблюдателя


Коэффициент пропорциональности Kp	Собственные числа матрицы динамики наблюдателя	Перерегулирование позиции тележки	Время регулирования позиции тележки	Максимальная амплитуда углового положения маятника
1				
2				
3				
5				
10				
15				
20				
30				

Запустите модель `lab4_modelSUPendulum` с наилучшими вариантами собственных чисел матриц динамики замкнутой системы и наблюдателя.

Запустите файл `lab4_plot1.m` для отображения на комплексной плоскости исходных и желаемых собственных чисел матриц динамики замкнутой системы и наблюдателя, построения графиков заданного и фактического положения тележки, скорости тележки, угла маятника. Скопируйте в отчет полученные графики с помощью команды меню `Edit\Copy Figure`.

4.3.3. Порядок запуска системы управления в режиме реального времени

Внимательно изучите последовательность запуска системы управления в режиме реального времени. Неправильный порядок запуска может привести к повреждению оборудования.

- Удостоверьтесь, что питание блока управления включено (выключатель слева от кнопки START подсвечен). Зеленая кнопка START не должна быть включена. Если она включена (горит), нажмите кнопку STOP.
- Переместите тележку в нулевую позицию и зафиксируйте маятник в нижнем положении, чтобы он не раскачивался.
- Перед запуском системы в режиме реального времени и после каждой коррекции mdl-файла необходимо скомпилировать алгоритм управления, для этого в меню Tools\Real-Time Workshop выбрать пункт Build model.
- После появления в командном окне сообщения «Successful completion...» соедините программу с объектом, нажав кнопку Connect to Target . Решение еще не будет запущено, но интерфейсная карта PCI-1711 будет правильно инициализирована.
- Убедитесь, что вокруг движущихся частей стенда нет посторонних предметов, мешающих его движению.
- Будьте готовы в любое время нажать кнопку STOP, чтобы не вывести стенд из строя при некорректной работе программы.
- Нажмите зеленую кнопку START на блоке управления маятником.
- Запустите решение, нажав кнопку Run в среде SIMULINK.
- После окончания решения нажмите кнопку STOP.

Соблюдайте указанные меры предосторожности при каждом запуске программы в реальном времени.

4.3.4. Проверка результатов модального синтеза управления инвертированным маятником на стенде

Для проверки результатов модального синтеза на стенде откройте файл lab4_SUPendulum.mdl (рис. 4.3). В данном файле для управления стендом в режиме инвертированного маятника используется модальный регулятор, настроенный в разделе 4.3.2. Блок «Согласование с аппаратурой» используется для преобразования выходного сигнала регулятора в диапазоне от -24 В до 24 В в диапазон от $-2,5$ В до $2,5$ В. Исходные данные для работы с файлом lab4_SUPendulum.mdl находятся в m-файле lab4_ModelSUPendulum_ini.m.

Выходные данные записываются в рабочую область MATLAB с помощью блока output_data. Данные в этом блоке записываются в виде структуры со временем (Structure with time) и используются для вывода графиков с помощью файла lab4_plot2.m.

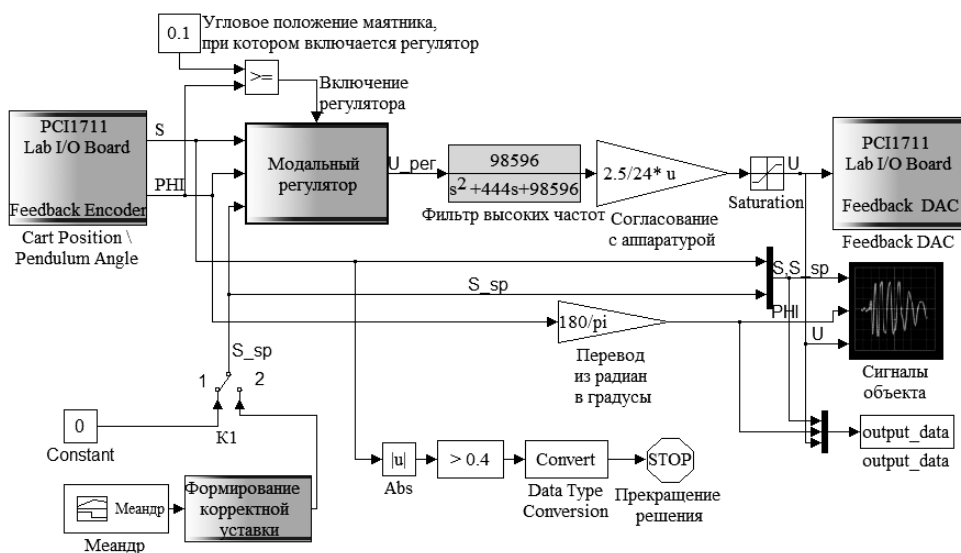


Рис. 4.3. Файл lab4_SUPendulum.mdl

Уставка формируется в виде биполярного меандра с периодом 20 с и амплитудой 0,3 В. Меандр начинает формироваться через 10 с после начала моделирования. В блоке «Формирование корректной уставки» (см. рис. 4.2) ограничивается скорость нарастания, и сглаживаются фронты сигнала.

При работе модального регулятора на стенде в режиме реального времени могут возникнуть высокочастотные колебания управляющего сигнала регулятора. Для обработки выходного сигнала регулятора с целью устранения высокочастотных колебаний используется фильтр высоких частот.

Запуск модального регулятора происходит при угловом положении маятника $\geq 0,1$ рад.

Для дополнительной безопасности при работе с файлом lab4_SUPendulum.mdl предусмотрено прерывание решения, если тележка отклонится от нулевого положения более чем на 0,4 м.

Перед запуском стенда внимательно изучите порядок запуска системы управления в режиме реального времени (раздел 4.3.3).

Перед запуском файла lab4_SUPendulum.mdl в присутствии преподавателя выполните следующие действия:

- 1) в m-файле lab4_ModelSUPendulum_ini.m. укажите желаемые полюса замкнутой системы и собственные числа матрицы динамики наблюдателя, выбранные в разделе 4.3.2, и сохраните файл;
- 2) в файле lab4_SUPendulum.mdl установите ключ K1 в положение «1». **Запрещается запускать решение, если ключ K1 установлен в положение «2»;**
- 3) скомпилируйте файл lab4_SUPendulum.mdl и убедитесь в отсутствии ошибок в процессе компиляции;

- 4) **неправильная работа алгоритмов может привести к резкому перемещению тележки и повреждению стенда. Запуск решения должен выполнить преподаватель. После запуска решения преподаватель должен аккуратно повернуть маятник на угол $\geq 0,1$ рад. При этом включится регулятор и произойдет стабилизация маятника в верхнем вертикальном положении;**
- 5) переключите ключ К1 в положение «2» и внимательно следите за обработкой входного сигнала.

После окончания решения с помощью файла `lab4_plot2.m` сохраните графики заданного и фактического положения тележки, углового положения маятника, управляющего сигнала. Сохраните графики для отчета.

Сделайте выводы о соответствии результатов моделирования работе стенда в реальном времени в режиме инвертированного маятника.

4.4. Содержание отчета

1. Исходные данные лабораторной работы: название работы, цель работы, параметры объекта управления, указанные в табл. 4.1, 4.2.
2. Процесс выбора желаемых собственных чисел матриц динамики замкнутой системы и наблюдателя (табл. 4.3, 4.4). Для наилучшего варианта приведите графики исходных и заданных собственных чисел матриц динамики замкнутой системы и наблюдателя, заданного и фактического положения тележки, скорости тележки, углового положения маятника.
3. Результаты работы модального регулятора на стенде в режиме инвертированного маятника. Графики заданного и фактического положения тележки, углового положения маятника, управляющего сигнала. Выводы о качестве работы замкнутой системы с модальным регулятором и соответствии результатов моделирования работе стенда в реальном времени.
4. Отчет оформляется на листах формата А4, допускается рукописное, печатное или комбинированное оформление. Листы отчета должны быть скреплены.

Контрольные вопросы

1. Поясните состав лабораторного стенда и взаимодействие его основных частей.

2. Укажите порядок запуска системы управления в режиме реального времени.
3. Поясните алгоритм получения векторно-матричного описания линеаризованной системы четвертого порядка.
4. Поясните процесс выбора желаемых собственных чисел матриц динамики замкнутой системы и наблюдателя.
5. Поясните состав модального регулятора и модели lab4_ModelSUPendulum.mdl.
6. Поясните состав и работу алгоритмов в файле lab4_SUPendulum.mdl.

Библиографический список

1. Говорухин В. Н. Компьютер в математическом исследовании / В. Н. Говорухин, Б. Г. Цибулин. СПб. : Питер, 2001. 624 с.
2. Малов А. В. Исследование линейных систем : метод. указания / А. В. Малов, Е. Э. Страшинин, А. В. Цветков. Екатеринбург : УГТУ-УПИ, 2005. 52 с.
3. Методы классической и современной теории управления : учебник : в 5 т. / Под ред. К. А. Пупкова и Н. Д. Егупова. М. : Издательство МГУ им. Н. Э. Баумана. Т. 3 : Синтез регуляторов систем автоматического управления, 2004. 616 с.
4. Поляков К. Ю. Теория автоматического управления для «чайников» : учеб. пособие. Часть 1 / К. Ю. Поляков. 2008. 80 с. Режим доступа: <http://kpolyakov.narod.ru/uni/teapot.htm>
5. Тарг С. М. Краткий курс теоретической механики: учеб. для вузов / С. М. Тарг. М. : Высшая школа, 1986. — 416 с.

Приложение 1

Тексты m-файлов, используемых в лабораторной работе № 1

Текст m-файла lab1.m

```
% Работа с векторно-матричными уравнениями линеаризованной системы

% вариант № (укажите № варианта)
clc% очистка экрана
clear all% очистка памяти
% объявление символьных переменных
syms m M L g J b d Kt Kb Ra La Rk b_vf
% укажите select=0 для работы с векторно-матричными уравнениями системы,
% линеаризованной в верхнем вертикальном положении
% или select=1 для работы с векторно-матричными
% уравнениями системы, линеаризованной
% в нижнем вертикальном положении
select=1;
switch select
case 0,
% укажите элементы матрицы A векторно-матричных
% уравнений системы, линеаризованной в верхнем
% вертикальном положении
a11 =; a12 =; a13 =; a14 =; a15 =;
a21 =; a22 =; a23 =; a24 =; a25 =;
a31 =; a32 =; a33 =; a34 =; a35 =;
a41 =; a42 =; a43 =; a44 =; a45 =;
a51 =; a52 =; a53 =; a54 =; a55 =;
case 1,
% укажите элементы матрицы A векторно-матричных
% уравнений системы, линеаризованной в нижнем вертикальном% положении
a11 =; a12 =; a13 =; a14 =; a15 =;
a21 =; a22 =; a23 =; a24 =; a25 =;
a31 =; a32 =; a33 =; a34 =; a35 =;
a41 =; a42 =; a43 =; a44 =; a45 =;
a51 =; a52 =; a53 =; a54 =; a55 =;
end
```

```

% укажите элементы матрицы B
b1 =; b2 =; b3 =; b4 =; b5 =;
% формирование матриц A, B, C, D
A = [a11 a12 a13 a14 a15; a21 a22 a23 a24 a25;...
a31 a32 a33 a34 a35; a41 a42 a43 a44 a45; a51 a52 a53 a54 a55];
B = [b1 b2 b3 b4 b5]';
C = [0 1 0 0 0; 0 0 0 1 0];
D = [0; 0];
display('Матрицы A, B, C, D в символьной форме')
% вывод матриц A, B, C, D в символьной форме в рабочую область
pretty(A)
pretty(B)
C
D
digits(5)% установка количества знаков после запятой
% присваивание переменным численных значений
m = sym(0.174, 'd');
M = sym(1.321, 'd');
L = sym(0.38, 'd');
g = sym(9.81, 'd');
J = sym(0.00099, 'd');
b = sym(0.05, 'd');
b_vf = sym(0.00568, 'd');
d = sym(0.005, 'd');
Kt = sym(0.05, 'd');
Kb = sym(0.05, 'd');
Ra = sym(2.5, 'd');
La = sym(0.0025, 'd');
Rk = sym(0.025, 'd');
% подстановка значений в матрицы
A=subs(A);
B=subs(B);
display('Матрицы A, B, C, D с подставленными значениями')
pretty(A)
pretty(B)
% Описание системы в пространстве состояний
sysOb = ss(double(A), double(B), C, D);
display('Передаточная функция')
tfsysOb = tf(sysOb); % вычисление передаточной функции
tfsysOb(1) % вывод передаточной функции объекта от u к S
tfsysOb(2) % вывод передаточной функции объекта от u к fi
% вычисление полюсов передаточной функции
display('Полюса передаточной функции')
% полюса передаточной функции объекта от u к S
pole(tfsysOb(1))
% полюса передаточной функции объекта от u к fi

```

```

pole (tfsysOb (2))
% вычисление нулей передаточной функции
display ('Нули передаточной функции')
zero (tfsysOb (1))% нули передаточной функции объекта от u к S
zero (tfsysOb (2))% нули передаточной функции объекта от u к fi
% вычисление собственных чисел матрицы динамики A
display ('Собственные числа матрицы A')
eig (double (A))
% построение графиков с помощью ltiview
ltiview ('step', tfsysOb)
ltiview ('impulse', tfsysOb)
ltiview ('bode', tfsysOb)
ltiview ('nyquist', tfsysOb)

```

Текст m-файла lab1_model1_params.m

```

% параметры модели lab1_model1.mdl

m = 0.174;
M = 1.321;
L = 0.38;
g = 9.81;
J = 0.00099;
Ra=2.5;
La = 0.0025;
b = 0.05;
d = 0.005;
Kb = 0.05;
Kt = 0.05;
Rk = 0.025;
b_vf = 0.00568;
% начальное угловое положения маятника
phi=0;

```

Текст m-файла lab1_model2_params.m

```

% параметры модели lab1_model2.mdl

m = 0.174;
M = 1.321;
L = 0.38;
g = 9.81;
J = 0.00099;
Ra=2.5;
La = 0.0025;
b = 0.05;

```



```

d = 0.005;
Kb = 0.05;
Kt = 0.05;
Rk = 0.025;
b_vf = 0.00568;
% отклонение от начального углового положения маятника
dphi=60;
% вычисление начального углового положения маятника в
% радианах
phi0=pi+dphi*pi/180;
% формирование вектора начальных условий для
% линеаризованной модели
X0 = [0 0 0 (-dphi) *pi/180 0];
% укажите элементы матрицы A векторно-матричных
% уравнений системы, линеаризованной в нижнем вертикальном
% положении
a11 =; a12 =; a13 =; a14 =; a15 =;
a21 =; a22 =; a23 =; a24 =; a25 =;
a31 =; a32 =; a33 =; a34 =; a35 =;
a41 =; a42 =; a43 =; a44 =; a45 =;
a51 =; a52 =; a53 =; a54 =; a55 =;

```

Текст m-файла lab1_plot.m

```

% вывод графиков положения тележки и углового положения маятника для не-
линейной и линеаризованной моделей
% голубым цветом показаны графики для нелинейной модели,
% красным — для линеаризованной
figure (1)% позиция тележки
line (position.time, position.signals.values,'Color','b','LineWidth',2);
line (positionL.time, positionL.signals.values,'Color','r','LineWidth',2);
grid on
hold on
xlabel ('Время, сек');
ylabel ('S, м');
legend ('Позиция тележки для нелинейной модели','Позиция тележки для лине-
аризованной модели');
figure (2)% угловое положения маятника
line (angle.time, angle.signals.values,'Color','b','LineWidth',2);
line (angleL.time, angleL.signals.values,'Color','r','LineWidth',2);
grid on
hold on
xlabel ('Время, сек');
ylabel ('\phi, град');
legend ('Угловое положение маятника для нелинейной модели','Угловое поло-
жение маятника для линеаризованной модели');

```

Приложение 2

Тексты m-файлов, используемых в лабораторной работе № 2

Текст m-файла lab2_params.m

```
% параметры объекта управления для модели  
% lab2_ident_model.mdl
```

```
m = 0.174;  
M = 1.321;  
L = 0.38;  
g = 9.81;  
J = 0.00099;  
Ra=2.5;  
La = 0.0025;  
b = 0.05;  
b_vf = 0.00568;  
d = 0.005;  
Kb = 0.05;  
Kt = 0.05;  
Rk = 0.025;
```

Текст m-файла lab2_plot1.m

```
% построение графиков позиции тележки, углового положения  
% маятника и входного напряжения двигателя
```

```
% построение графика позиции тележки  
figure (1)  
xlabel ('t, с');  
ylabel ('S, м');  
hold on;  
grid on;  
time=0.05:0.05:100;  
axis ([0 100 -0.15 0.15]);  
plot (time, ident_data (:,1), 'Color', 'red', 'LineWidth', 2);
```

```

legend ('Позиция тележки');
% построение графика углового положения маятника
figure (2)
xlabel ('t, с');
ylabel ('\phi, град');
hold on;
grid on;
time=0.05:0.05:100;
axis ([0 100 120 240]);
plot (time, ident_data (:,2),'Color','green','LineWidth',2);
legend ('Угловое положение маятника');
% построение графика управляющего напряжения
figure (3)
xlabel ('t, с');
ylabel ('U, В');
hold on;
grid on;
time=0.05:0.05:100;
axis ([0 100 -0.5 0.5]);
plot (time, ident_data (:,3),'Color','blue','LineWidth',2);
legend ('Управляющее напряжение');

```

Текст m-файла lab2_plot2.m

```

% построение графиков позиции тележки, углового положения
% маятника и входного напряжения двигателя с указанием
% длины массива данных

```

```

length_mass=; % укажите длину массива S
figure (1)% построение графика позиции тележки
xlabel ('t, с');
ylabel ('S, м');
hold on;
grid on;
time=0:0.05: (length_mass-1) *0.05;
axis ([0 length_mass*0.05 -0.15 0.15]);
plot (time, S,'Color','red','LineWidth',2);
legend ('Позиция тележки');
% построение графика углового положения маятника
figure (2)
xlabel ('t, с');
ylabel ('\phi, град');
hold on;
grid on;
time=0:0.05: (length_mass-1) *0.05;
axis ([0 length_mass*0.05 120 240]);

```

```
plot (time, PHI,'Color','green','LineWidth',2);
legend ('Угловое положение маятника');
% построение графика управляющего напряжения
figure (3)
xlabel ('t, с');
ylabel ('U, В');
hold on;
grid on;
time=0:0.05: (length_mass-1) *0.05;
axis ([0 length_mass*0.05-0.5 0.5]);
plot (time, U,'Color','blue','LineWidth',2);
time=begin_f*0.05+0.05:0.05: end_f*0.05;
axis ([begin_f*0.05+0.05 end_f*0.05-0.5 0.5]);
plot (time, U,'Color','blue','LineWidth',2);
legend ('Управляющее напряжение');
```

Приложение 3

Тексты m-файлов, используемых в лабораторной работе № 3

Текст m-файла lab3_ModelSUCrane_ini.m

% исходные данные для работы файлов lab3_ModelSUCrane.mdl
%и lab3_SUCrane.mdl

```
clc
clear all
% параметры объекта управления
m = 0.174;
M = 1.321;
L = 0.38;
g = 9.81;
J =;% укажите уточненное значение параметра, полученное в
% лабораторной работе № 2.
Ra=2.5;
La = 0.0025;
b =0.05;
d =;% укажите уточненное значение параметра, полученное в
% лабораторной работе № 2.
Kb = 0.05;
Kt = 0.05;
Rk = 0.025;
b_vf=;% укажите уточненное значение параметра, полученное в
% лабораторной работе № 2.
phi = pi;
% укажите элементы матриц линеаризованной модели 4-го
% порядка
A4_11 =; A4_12 =; A4_13 =; A4_14 =;
A4_21 =; A4_22 =; A4_23 =; A4_24 =;
A4_31 =; A4_32 =; A4_33 =; A4_34 =;
A4_41 =; A4_42 =; A4_43 =; A4_44 =;
B4_1 =; B4_2 =; B4_3 =; B4_4 =;
A4_plant_Cart = [A4_11 A4_12 A4_13 A4_14; A4_21 A4_22
```

```

A4_23 A4_24; A4_31 A4_32 A4_33 A4_34; A4_41 A4_42 A4_43
A4_44];
% собственные числа матрицы динамики A
eigA4Cart = eig (A4_plant_Cart)
B4_plant_Cart = [B4_1 B4_2 B4_3 B4_4]';
C4_plant_Cart = [1 0 0 0; 0 0 1 0];
D4_plant_Cart = [0; 0];
% формирование ss-модели линеаризованной системы 4-го
% порядка
sys1_r = ss (A4_plant_Cart, B4_plant_Cart, C4_plant_Cart,
D4_plant_Cart);
W_s1_r = tf (sys1_r);
% Вычисление нулей и полюсов ПФ разомкнутой системы
[zeros_s1_r1, poles_s1_r1, koef_s1_r1] = zpkdata (sys1_r (1), 'v');
disp ('Нули и полюсы ПФ разомкнутой системы по каналу S');
zeros_s1_r1
poles_s1_r1
[zeros_s1_r2, poles_s1_r2, koef_s1_r2] = zpkdata (sys1_r (2), 'v');
disp ('Нули и полюсы ПФ разомкнутой системы по каналу phi');
zeros_s1_r2
poles_s1_r2

```

Текст m-файла lab3_plot1.m

```

% вывод результатов моделирования

% Закрытие всех предыдущих рисунков
set (0, 'ShowHiddenHandles', 'on')
delete (get (0, 'Children'))
close all
figure (1)
xlabel ('Re');
ylabel ('Im');
plot (real (poles_s1_r1), imag (poles_s1_r1), 'rs', 'Color', 'b');
hold on
plot (real (zeros_s1_r1), imag (zeros_s1_r1), 'ro', 'Color', 'r');
grid on
legend ('Полюсы линеаризованной системы 4-го порядка', 'Нули
линеаризованной системы 4-го порядка')
figure (2)
xlabel ('Re');
ylabel ('Im');
plot (real (P4_SuCart), imag (P4_SuCart), 'rs', 'Color', 'b');
hold on
plot (real (P4_nab), imag (P4_nab), 'rd', 'Color', 'r');
grid on

```

```

legend ('Желаемые собственные числа матрицы динамики замкнутой системы',
собственные числа матрицы динамики наблюдателя');
figure (3)
ylabel ('S, S3, dS/dt, м, м/с');
xlabel ('t, с');
line (output_data.time, output_data.signals.values (:,1), 'Color', 'b', 'LineWidth', 2);
line (output_data.time, output_data.signals.values (:,2), 'Color', 'r', 'LineWidth', 2);
line (output_data.time, output_data.signals.values (:,3), 'Color', 'g', 'LineWidth', 2);
grid on
hold on
legend ('Позиция тележки', 'Заданная позиция тележки', 'Скорость тележки');
figure (4)
ylabel ('/phi, град');
xlabel ('t, с');
line (output_data.time, output_data.signals.values (:,4), 'Color', 'b', 'LineWidth', 2);
hold on
grid on
legend ('Угловое положение маятника');
figure (5)
ylabel ('U, В');
xlabel ('t, с');
line (output_data.time, output_data.signals.values (:,5), 'Color', 'r', 'LineWidth', 2);
grid on
hold on
legend ('Управляющий сигнал');

```

Текст m-файла lab3_plot2.m

```

% вывод результатов работы системы в режиме реального
% времени

% Закрытие всех предыдущих рисунков
set (0, 'ShowHiddenHandles', 'on')
delete (get (0, 'Children'))
close all
figure (1)
xlabel ('Re');
ylabel ('Im');
plot (real (poles_sl_r1), imag (poles_sl_r1), 'rs', 'Color', 'b');
hold on
plot (real (zeros_sl_r1), imag (zeros_sl_r1), 'ro', 'Color', 'r');
grid on
legend ('Полюса линеаризованной системы 4-го порядка', 'Нули
линеаризованной системы 4-го порядка')
figure (2)
xlabel ('Re');

```

```

ylabel ('Im');
plot (real (P4_SuCart), imag (P4_SuCart), 'rs', 'Color', 'b');
hold on
plot (real (P4nab), imag (P4nab), 'rd', 'Color', 'r');
grid on
legend ('Желаемые собственные числа матрицы динамики замкнутой системы',
собственные числа матрицы динамики наблюдателя');
figure (3)
ylabel ('S, S3, dS/dt, м, м/с');
xlabel ('t, с');
line (output_data.time, output_data.signals.values (:,1), 'Color', 'b', 'LineWidth', 2);
line (output_data.time, output_data.signals.values (:,2), 'Color', 'r', 'LineWidth', 2);
grid on
hold on
legend ('Позиция тележки', 'Заданная позиция тележки');
figure (4)
ylabel ('/phi, град');
xlabel ('t, с');
line (output_data.time, output_data.signals.values (:,3), 'Color', 'b', 'LineWidth', 2);
hold on
legend ('Угловое положение маятника');
figure (5)
ylabel ('U, В');
xlabel ('t, с');
line (output_data.time, output_data.signals.values (:,4), 'Color', 'r', 'LineWidth', 2);
grid on
hold on
legend ('Управляющий сигнал');

```


Приложение 4

Тексты m-файлов, используемых в лабораторной работе № 4

Текст m-файла lab4_ModelSUCrane_ini.m

```
% исходные данные для работы файлов
% lab4_ModelSUPendulum.mdl и lab4_SUPendulum.mdl
clc
clear all
% параметры объекта управления
m = 0.174;
M = 1.321;
L = 0.38;
g = 9.81;
J =;% укажите уточненное значение параметра, полученное в
% лабораторной работе № 2.
Ra=2.5;
La = 0.0025;
b =;% укажите уточненное значение параметра, полученное в
% лабораторной работе № 2
d =;% укажите уточненное значение параметра, полученное в
% лабораторной работе № 2.
Kb = 0.05;
Kt = 0.05;
Rk = 0.025;
b_vf=;% укажите уточненное значение параметра, полученное в
% лабораторной работе № 2.
phi = pi;
% укажите элементы матриц линеаризованной модели
% четвертого порядка
A4_11 =; A4_12 =; A4_13 =; A4_14 =;
A4_21 =; A4_22 =; A4_23 =; A4_24 =;
A4_31 =; A4_32 =; A4_33 =; A4_34 =;
```

```

A4_41 =; A4_42 =; A4_43 =; A4_44 =;
B4_1 =; B4_2 =; B4_3 =; B4_4 =;
A4_Pendulum = [A4_11 A4_12 A4_13 A4_14; A4_21 A4_22 A4_23
A4_24; A4_31 A4_32 A4_33 A4_34; A4_41 A4_42 A4_43
A4_44];
eigA4_Pendulum = eig (A4_Pendulum);
B4_Pendulum = [B4_1 B4_2 B4_3 B4_4]';
C4_Pendulum = [1 0 0 0; 0 0 1 0];
D4_Pendulum = [0; 0];
% формирование ss-модели линеаризованной системы
% четвертого порядка
sys1_r = ss (A4_Pendulum, B4_Pendulum, C4_Pendulum, D4_Pendulum);
W_s1_r = tf (sys1_r);
% Вычисление нулей и полюсов ПФ разомкнутой системы
[zeros_s1_r1, poles_s1_r1, koef_s1_r1] = zpndata (sys1_r (1), 'v');
disp ('Нули и полюсы ПФ разомкнутой системы по каналу S');
zeros_s1_r1
poles_s1_r1
[zeros_s1_r2, poles_s1_r2, koef_s1_r2] = zpndata (sys1_r (2), 'v');
disp ('Нули и полюсы ПФ разомкнутой системы по каналу phi');
zeros_s1_r2
poles_s1_r2

```

Тексты m-файлов lab4_plot1.m и lab4_plot1.m аналогичны m-файлам lab3_plot1.m и lab3_plot2.m (см. приложение к лабораторной работе № 3).

Учебное издание

Зиновик Алексей Георгиевич, **Страшинин** Евгений Эрастович

**Исследование и синтез системы управления стэнда
«Тележка с подвижным маятником»**

Редактор *О. В. Климова*
Корректор *А. А. Загоруйко*
Верстка *О. П. Игнатъевой*

Подписано в печать 01.10.2014. Формат 70×100 1/16.
Бумага писчая. Плоская печать. Усл. печ. л. 6,77.
Уч.-изд. л. 5,6. Тираж 50 экз. Заказ 1631.

Издательство Уральского университета ИПЦ УрФУ
620049, Екатеринбург, ул. С. Ковалевской, 5
Тел.: (343) 375-48-25, 375-46-85, 374-19-41
E-mail: rio@ustu.ru

ОБ АВТОРАХ



ЗИНОВИК

Алексей Георгиевич

выпускник кафедры Автоматики
Института Радиоэлектроники и ин-
формационных технологий УРФУ



СТРАШИНИН

Евгений Эрастович

канд. техн. наук, доцент, профес-
сор кафедры «Автоматика». Ав-
тор более 190 научных и учебно-
методических работ. Область
научных интересов — теория
и практика управления динами-
ческими объектами